Reasonable Basic Calculus

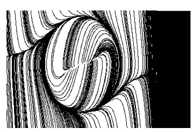
œ

Alain Schremmer

Reasonable Basic Calculus

According To The Real World
If Only Because
Signed *Decimal* Numbers Are The *Real* "Real Numbers"[1]

Even if the real *real world isn't always reasonable!*



FreeMathTexts.org
Free as in free beer. Free as in free speech.

Version 2.4 — Wednesday 15th November, 2023, 23:20

[1] Clicking on *anything* in reddish characters will get you within at most a short scrolling distance from the relevant place and just hovering will show you a picture thereof.

To Françoise.

For bringing out the song[2] *in the Well-Tempered Clavier,*
For growing so many, so many trees,
For being, among others, a wonderful cuisinière[3] *and a great knitter,*
And, neither last nor least, for being a real *mathematician:* BOUNDARY VALUE PROBLEMS *"arise in several branches of physics"*[4].

[2] Professional pianists have to exhibit their *technique* before anything else.
[3] French chefs have often been accused of stealing from cuisinières.
[4] https://en.wikipedia.org/wiki/Boundary_value_problem.

Contents

What is important is the **real world**, that is physics, but it can be explained only in mathematical terms.[5]

Dennis Serre[6]

real world
RBC
Underwood Dudley

Preface You *Don't* Have To Read

The prefaces of standard calculus books are never for you*, the reader, but nearly always to convince* teachers *that the book is exactly what they want their students to* buy.

But this Preface You Don't *Have To Read is the preface of a* free text *and thus* is *for* you*, the reader.*

Authors of calculus books invariably claim that *their* book is different. And of course so does the author of this REASONABLE BASIC CALCULUS, ***RBC*** for short! But in exactly what way(s)? First, though, how about standard books?

1 For Whom Standard Books Toll

Back in 1988, **Underwood Dudley**[7] published in the American Mathematical Monthly a wonderful article about calculus books—camouflaged as a Book Review![8]— which he said he wrote after having "*examined 85 separate and distinct calculus books.*"[9]

[5]Bulletin of the American Mathematical Society, Vol 47 Number 1 Pages 139-144

[6]https://en.wikipedia.org/wiki/Denis_Serre.

[7]https://en.wikipedia.org/wiki/Underwood_Dudley

[8]Here is the actual review in its entirety: '*The book by Simmons is a fine one. It was written with care and intelligence. It has good problems, and the historical material is almost a course in the history of mathematics. It is nicely printed, well bound, and expensive. Future historians of mathematics will look back on it and say,* 'Yes, that is an excellent example of a late twentieth-century calculus book.'").

[9]https://www.maa.org/sites/default/files/0002989051112.di991736.99p03667.pdf

Loomis
L'Hospital's Rule
pathological
Silvanus Thompson
Hung-Hsi Wu

Dudley's first point was that "*Calculus books should be written for students*". As an example of one such, Dudley gives **Elias Loomis**'[10] *Elements of the Differential and Integral Calculus*[11] from 1851. He points out that Loomis' "*proof of **L'Hospital's Rule** was short, simple, and clear, and also one which does not appear in modern texts because it fails for certain **pathological** examples*[12]".

*E.g. G. Strang in his Calculus (p151): "*I regard the discussion below as optional in a calculus course *(but required in a calculus book)."*

A bit later, Dudley continues: "*It is a still better idea to strive for clarity and let students see what is really going on, which is what Loomis did, rather than putting 'rigor' first. But nowadays, authors cannot do that. They must protect against some colleague snootily writing to the publisher "Evidently Professor Blank is unaware that his so-called proof of L'Hospital Rule is faulty, as the following well-known example shows. I could not possibly adopt a text with such a serious error."*'

At less than $10!

Which, these days, would be an unspeakable horror!

As another example of a book written for students, Dudley gives **Silvanus Thompson**'s[13] *Calculus Made Easy*[14] from 1910 which was very successful and is in fact still in print. Dudley is visibly enchanted to report that "*Chapter 1, whose title is '*To Deliver You From The Preliminary Terrors*' forthrightly says that* dx *means 'a little bit of* x". (Significantly enough, Thompson was a professor of physics and an electrical engineer.)

Another important point Dudley made was that "*First-semester calculus has no application.*" Of course there is no question about CALCULUS being about the Real World. Absolutely none. The only thing is, the Real World is in the eye of the beholder and the beholder usually is, here again, the teacher. And so, of course, Dudley riffs on "*Applications being so phony*".

Well, RBC sure wasn't!

Dudley conclusion was that "*It is a shame, and probably inevitable that calculus books are written for calculus teachers.*"

And, indeed, as Dudley predicted, nothing has changed to this day.

In fact, twenty-seven years later, and even though it was about "*school math*", **Hung-Hsi Wu**[15] responded in the Notices of the American Mathematical Society to Elizabeth Green's New York Times article *Why Do Americans Stink at Math?*[16] in these terms:

[10] https://en.wikipedia.org/wiki/Elias_Loomis
[11] Free at https://archive.org/details/elementsofdiffer00loom/page/n4/mode/2up
[12] https://en.wikipedia.org/wiki/Pathological_(mathematics)#Pathological_examples
[13] https://en.wikipedia.org/wiki/Silvanus_P._Thompson
[14] Free at https://archive.org/details/CalculusMadeEasy/page/n4/mode/2up.
[15] https://math.berkeley.edu/~wu/
[16] https://www.nytimes.com/2014/07/27/magazine/why-do-americans-stink-at-math.html

"*If Americans do "stink" at math, clearly it is because they find the math in school to be unlearnable.* [...] *For the past four decades or so the mathematics contained in standard textbooks has played havoc with the teaching and learning of school mathematics.*"[17]

Da Vinci
develop
use

2 For Whom *This* Text?

In other words, RBC is for people allergic to just being "shown how to do it", for people who like lo look under the hood and even to reinvent the wheel to see what makes it turn . . .

The short answer is that, inasmuch as, in the words of **Leonardo da Vinci** (1452-1519), "*Learning is the only thing the mind never exhausts, never fears, and never regrets.*"[18], *RBC* wants to let people who like to read, ponder, wonder, . . . **develop** a CALCULUS they can **use** in the real world.

EXAMPLE 0.1. *RBC* begins with Reasonable Numbers, a "zeroth" chapter on aspects of numbers that are basic to real world calculations but very rarely discussed in ARITHMETIC textbooks.
Then, to introduce the reader to functions, which are to CALCULUS what numbers are to ARITHMETIC, *RBC* continues with Part I which, following Da Vinci, starts with Relations Given By Data namely, as in the experimental sciences, given by way of lists, tables, and plots, and continues with Functions Given Graphically and The Looks Of Functions.
Only then, in Part II, does CALCULUS proper begin with the introduction of Global Input-Output Rules which are the simplest way to give functions that can be *calculated* with.

The reason for so many pages is so many pictures, so many **EXAMPLES** *and so many* **DEMOS**.

And, by the way, *RBC* is completely self contained:

• Just in case you missed the subtitle of the book: if you can compare/add/subtract/multiply/divide *signed decimal numbers*, you need not worry about being "prepared".

And even if you can't, Dealing With Decimal Numbers (Appendix A, Page 225) will always be a mere click away.

• The URLs in the footnotes are just references—mostly to articles in Wikepedia[19]— to help people curious to know *more* about the matter at hand.

But even if this short answer may *look* good, it surely doesn't *say* very much and what follows are progressively longer answers for those who, before deciding whether or not to get into something, want to know *more precisely* what it is they would be getting into and *why* they would want to do that in the first place.

On the other hand, should you prefer to go and see for yourself, clicking on anything in redish characters, for instance in EXAMPLE 0.1 - For Whom This *Text? (Page xiii). will get you there.*

[17] http://www.ams.org/notices/201505/rnoti-p508.pdf
[18] https://www.azquotes.com/author/15101-Leonardo_da_Vinci
[19] https://en.wikipedia.org/wiki/Main_Page

Einar Hille
George Sarton
John Holt

3 The Three Things This Text Wants To Do

This not-so-short answer begins with the fact that, for the exact same reason Hung-Hsi Wu gave for why "*Americans 'stink' at math*", it can be maintained that so-called Math Anxiety invariably originates with the standard text-books, in the best cases because the book leaves so much going without saying that reason has become all but invisible, in the worst cases because the book has been gutted down to the disconnected "*facts and skills*" deemed necessary to pass some exam so that no reason remains at all.

As your Doctor will tell you, they had to take one year of Calculus in college to be able to apply to Medical School, but they can't remember a word of it.

In contrast, *RBC* wants to do three things:

- As **Einar Hille**[20] wrote, "*Mathematics is neither accounting nor the theory of relativity. Mathematics is much more than the sum total of its applicaations no matter how important and diversified these may be. It is* a way of thinking."[21] (Emphasis added.)
  Of course, a way of thinking cannot be taught or even described and can only be *learned* from *experience*. Fortunately, as **George Sarton**[22] wrote, "*It is only a matter of perseverance and of good will. Only thus will you acquire a* method *of thought. And if one cannot reproach anyone for being ignorant of this or that—for ignorance is not a sin—it is legitimate to reproach one with poor reasoning.* [...] [T]*his scientific sincerity is only achieved by the attentive study of a specific subject.*"[23]

  In other words, no "show and tell"*. Just* think*.*

  And so, the first thing *RBC* wants to do is to facilitate your "*attentive study*" of CALCULUS by presenting matters to you in a way that will make *reasonable* sense to *you*.

- As **John Holt**[24] wrote, "*Human beings are born intelligent. We are by nature question-asking, answer-making, problem-solving animals, and we are extremely good at it, above all when we are little. But under certain conditions, which may exist anywhere and certainly exist almost all of the time in almost all schools, we stop using our greatest intellectual powers, stop wanting to use them, even stop believing that we have them.*"[25]
  Which is why *RBC* does *not* have any **EXERCISE**: the important questions are those *you* will be wondering about. Of course, you would be quite

[20] https://en.wikipedia.org/wiki/Einar_Hille
[21] Einar Hille, *Analysis*, 1964
[22] https://en.wikipedia.org/wiki/George_Sarton
[23] As quoted from his letters by his daughter, May Sarton, in her book *I Knew a Phoenix*
[24] https://en.wikipedia.org/wiki/John_Holt_(educator)
[25] John Holt *How Children Fail* A classic, first published in the 60s. Free download from https://archive.org/download/HowChildrenFail/HCF.pdf

*In other words, no "*drill and test*". Just* experiment.

right to ask how you will know if you *have* learned CALCULUS but the answer still is: when you will have become able to answer most of *your* questions by *yourself.*

You probably won't like that *one one bit! (At least until you find out that you* can.)

And so, the second thing *RBC* wants to do is to present and discuss issues in a way that will enable you, one day, to look into some further aspects of CALCULUS all by yourself.

And, why not, even become *a mathematician.*

- As **Etienne Ghys**[26] wrote, "*I have now learned that precision and details are frequently necessary in mathematics, but I am still very fond of promenades.* [...] *You have to be prepared to get lost from time to time, like in many promenades.* [...] *You will have to accept half-baked definitions.* [...] *I am convinced that* **mathematical ideas** *and* **examples** *precede proofs and definitions.*"[27] (Emphasis added.)

  And so, the third thing *RBC* wants to do is to be a pleasant promenade for you.

Etienne Ghys
Newton
infinitesimal
limit
Leibniz
L'Hospital
Cauchy
Robinson
Arnold

4 Nano History Of Calculus

For the philosophically inclined, the history of how CALCULUS came about[28] can be fascinating but for those *just a tiny little bit curious*, here is probably the shortest possible version:

CALCULUS was created in the late 1600s, first by **Newton**[a], initially by way of **infinitesimals**[b] but eventually by way of **limits**[c], and, a bit later but completely *independently*, by **Leibniz**[d], by way of *infinitesimals*.

The first of the many editions of the first CALCULUS text, *Infinitesimal Calculus with Application to Curved Lines*, by **L'Hospital**[e], is from 1696.

Right away, all *scientists*, *engineers* and *mathematicians*—except British ones, presumably out of loyalty to Newton—started using *infinitesimals* routinely even though it was almost immediately realized that infinitesimals—as well as limits—were not rigorously defined. (Bishop Berkely even called them "*ghost of departed quantities*"[f].)

If only because 'limits' can't be computed *but only* guessed *and then* checked *to see if they* are *the 'limit'.*

And, even though, over a century later, most *mathematicians* switched to limits which had finally been rigorously defined by **Cauchy**[g], *scientists*, and for a long time even *differential geometers*, continued to use *infinitesimals*[h]

[26] https://en.wikipedia.org/wiki/%C3%89tienne_Ghys

[27] Etienne Ghys, *A singular mathematical promenade.* 2017. Free download from https://arxiv.org/abs/1612.06373

[28] https://en.wikipedia.org/wiki/History_of_calculus

hyperreal number
Fields Medal
Lagrange
polynomial approximation
decimal approximation
Henri Poincaré
asymptotic expansion
Poincaré expansion

Guess what: 'infinitesimals' are still avoided like the plague by most mathematicians not to mention—but that goes without saying—math teachers!

Of course, unlike Lagrange, RBC will not deal with pathological cases.

Most unfortunately, though, most teachers still confuse polynomial *approximations, which have only* so *many terms, with 'Taylor* series*' which have* infinitely *many terms and which RBC will stay away from.*

Then, in 1961, **Abraham Robinson**[i], three years over the age limit for the **Fields Medal**[j], finally succeeded in defining infinitesimals *rigorously* using the **hyperreal numbers**[k,l] that **Edwin Hewitt**[m] had pioneered in 1948.

Yet, as **Vladimir Arnold**[n]—a great mathetician who was prevented from getting the Fields Medal because of his public opposition to the persecution of dissidents in the Soviet Union during most of the 1970s and 1980s—wrote in 1990: "*Nowadays, when teaching analysis, it is not very popular to talk about infinitesimal quantities. Consequently present-day students are not fully in command of this language. Nevertheless, it is still necessary to have command of it.*"

On the other hand, a long time before all that, around 1800, **Lagrange**[o], one of the greatest mathematicians ever, who explicitly wanted to free CALCULUS from "*any consideration of infinitesimals, vanishing quantities, limits and fluxions*", had developed an approach by way of polynomial approximations, which are to CALCULUS what decimal approximations are to ARITHMETIC. And, even though, having realized that polynomial approximations could not deal with certain pathological cases, Lagrange had reverted to infinitesimals, polynomial approximations will be what *RBC* will employ.

In fact, beginning around 1880, yet another all time great mathematician, **(Henri) Poincaré**[p], had employed polynomial approximations to solve a very large number of problems so that Lagrange's polynomial approximations are now known as **Poincaré expansions** or **asymptotic expansions**[q].

[a] https://en.wikipedia.org/wiki/Isaac_Newton
[b] https://en.wikipedia.org/wiki/Infinitesimal
[c] https://en.wikipedia.org/wiki/Limit_(mathematics)
[d] https://en.wikipedia.org/wiki/Gottfried_Wilhelm_Leibniz
[e] https://en.wikipedia.org/wiki/Guillaume_de_l%27H%C3%B4pital
[f] https://en.wikipedia.org/wiki/The_Analyst#Ghosts_of_departed_quantities
[g] https://en.wikipedia.org/wiki/Augustin-Louis_Cauchy
[h] https://en.wikipedia.org/wiki/Calculus#Limits_and_infinitesimals
[i] https://en.wikipedia.org/wiki/Abraham_Robinson
[j] https://en.wikipedia.org/wiki/Fields_Medal)
[k] https://en.wikipedia.org/wiki/Hyperreal_number
[l] https://arxiv.org/pdf/2210.07958.pdf
[m] https://en.wikipedia.org/wiki/Edwin_Hewitt
[n] https://en.wikipedia.org/wiki/Vladimir_Arnold
[o] https://en.wikipedia.org/wiki/Joseph-Louis_Lagrange
[p] https://en.wikipedia.org/wiki/Henri_Poincar%C3%A9
[q] https://en.wikipedia.org/wiki/Asymptotic_expansion

5 The Paper World

paper world
word
noun
adjective
verb
object language
calculus language
calculus word
calculus noun
calculus adjective
calculus verb
metalanguage
ordinary English
ordinary English word
ordinary English noun
ordinary English adjective
calculus verb
Eugene Wigner
employ

The *long* answer begins with the fact that dealing with the real world, in the sciences as well as in the trades, requires a **paper world** involving *two* languages[14], each with its own **words**, **nouns**[15], **adjectives**[16] and **verbs**[17]:

A. An **object language** which in *RBC* will be the **calculus language** with its **calculus words**, namely **calculus nouns**, **calculus adjectives**, and **calculus verbs**,

Example 0.2. Carpenters have an object language that includes words such as ledger, purlin, riser, stringer, etc[a]

[a] https://www.mycarpentry.com/carpentry-terms.html

B. A **metalanguage** which in *RBC* will be **ordinary English** with its **ordinary English words**, namely **ordinary English nouns**, **ordinary English adjectives**, and **ordinary English verbs**.

Example 0.3. When the French author of *RBC* first learned English, ordinary English was his object language and French was his metalanguage.

Concerning the relevance of the paper world to the real world, here are two articles very much to the point:

- A very famous, if somewhat dense, article on "*The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics.*"[a] by **Eugene Wigner**[b],

which eventually started:

- A lively discussion on *natural law* and *mathematics* in Quanta Magazine[c]

[a] https://www.maths.ed.ac.uk/~v1ranick/papers/wigner.pdf
[b] https://en.wikipedia.org/wiki/Eugene_Wigner
[c] https://www.quantamagazine.org/puzzle-solution-natural-law-and-elegant-math-20200117/

In which the real world is invoked as often as here.

Language 0.1 *RBC* will distinguish between the calculus words use and **employ**. This may be overdoing things a bit but will help

[14] https://en.wikipedia.org/wiki/Metalanguage
[15] (https://en.wikipedia.org/wiki/Noun
[16] https://en.wikipedia.org/wiki/Adjective
[17] https://en.wikipedia.org/wiki/Verb

MODEL THEORY
grammar
sentence
calculus grammar
calculus sentence
ordinary English grammar
ordinary English sentence
syntactics

mark the difference between the two roles the reader will play in *RBC* namely:

- the reader learning to *develop* CALCULUS who, for instance and as we will see, will employ Generic given numbers (Subsection 3.2, Page 9) to keep things open,

and

- the reader learning to *use* CALCULUS who will use given numbers of *their* own choice.

This *explicit* distinction between real world and paper world is at the core of a relatively new part of MATHEMATICS called **MODEL THEORY**[18].

1. Syntactics The paper world should also include **grammars**[19] to assemble words into **sentences**[20] independently of any reference to the real world:

- A **calculus grammar** for assembling calculus words into **calculus sentences**.

That there be no *reference to the real world is* essential *to the grammar of* computer *languages.*

Since an understanding of the calculus grammar is necessary to develop CALCULUS, *RBC* will introduce the calculus grammatical rules as and when needed.

- An **ordinary English grammar** for assembling ordinary English words into **ordinary English sentences** but what grammar the reader learned in school will normally be quite enough to deal with the ordinary English of *RBC* and therefore

AGREEMENT 0.1 In *RBC*, ordinary English grammar will go completely without saying.

FOR THOSE INTERESTED: There is actually a lot more to languages than assembling words into sentences and **syntactics**[21] includes issues such as word order, grammatical relations, hierarchical sentence structures, etc

[18] https://plato.stanford.edu/entries/modeltheory-fo/
[19] https://en.wikipedia.org/wiki/Grammar
[20] https://en.wikipedia.org/wiki/Sentence_(linguistics)
[21] https://en.wikipedia.org/wiki/Syntax

2. Semantics. While *RBC* takes the knowledge of ordinary English for granted, the **semantics**[22] of the calculus language, that is "*The question of what is a proper basis for deciding how words, symbols, ideas and beliefs may properly be considered to truthfully refer to meaning.*"[23] must be discussed.

semantics
precise
mean
entity
refer
calculus statement
say
decide
truth value
TRUE
FALSE
Alfred Tarski
Aka "*real McCoy*".[24]

The difficulty is in the several ways in which the calculus language and the ordinary English language are inextricably tied.

i. The first way is how ordinary English is employed to make **precise** the **meaning** of ordinary English words then to be employed as calculus word by describing as precisely as possible the **entities** in the real world to be **refered** to.

Example 0.4. As Chief Inspector Kan reminds Inspector Van der Valk in *Criminal Conversation*[a], Nicolas Freeling[b]'s thriller, "*Law depends on the precise meaning of words*".

[a] Actually a legal term: https://en.wikipedia.org/wiki/Criminal_conversation
[b] https://en.wikipedia.org/wiki/Nicolas_Freeling

ii. Then, inasmuch as *RBC* will be restricted to **calculus statements**, that is to calculus sentences **saying** something *clear* and *precise*, ordinary English will be employed to **decide** the **truth value**[25] of calculus statements, that is whether the calculus statements are **TRUE** or **FALSE**, that is whether what the calculus statements say about the real world is or is not actually the case.

Example 0.5. Assume—just for the sake of this **Example**—that ordinary English is our *object* language.
Then, both "The moon is made of green cheese"[a] and "The moon is dreaming" are (grammatically correct) sentences in the object language but, while the sentence "The moon is made of green cheese" is a (FALSE) *statement*, what the sentence "The moon is dreaming" says is not clear so that the sentence "The moon is dreaming" is *not* a *statement* and thus neither TRUE nor FALSE.

[a] https://en.wikipedia.org/wiki/The_Moon_is_made_of_green_cheese

This model-theoretic view of truth is due to **Alfred Tarski** [26] who "[would

[22] https://en.wikipedia.org/wiki/Semantics
[23] https://en.wikipedia.org/wiki/Meaning_(philosophy)#Truth_and_meaning
[24] https://en.wikipedia.org/wiki/The_real_McCoy
[25] https://en.wikipedia.org/wiki/Semantic_theory_of_truth
[26] https://en.wikipedia.org/wiki/Alfred_Tarski

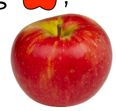
stand
model theory
explain

however not] claim [it was] the 'right' one [other than in *mathematics*]."

iii. Because the real world entities that the calculus words refer to cannot be *exhibited* in the paper world, the third way ordinary English words will be employed will be by **standing** in the paper world for real world entities. So, we will often employ the same ordinary English word both as a stand-in for a real world entity and as a paper world name for that entity.
However, other kinds of paper world stand-ins such as drawings, pictures, etc can be employed too.

Which can be eaten*.*

Example 0.6. Assume—just for the sake of this **Example**—that the *object* language is French. Then, the French word "pomme" is the word in the object language *refering* to the real world *entity* whose paper world *stand-in* could be any of:

- The ordinary English word "apple",
- The drawing ,
- The picture  .

None of which can be eaten*!*

Or, you may want to look up the author's A Model-Theoretic Introduction to Mathematics[27]*.*

Actually, it would be totally counterproductive.

However, while remaining aware of how essential the distinction between the object language and the metalanguage is, systematically distinguishing ordinary English words as calculus words from ordinary English words as stand-in word and refering to from standing-in for, would not serve any purpose in *RBC* . So:

> **Agreement 0.2** *RBC* will often use ordinary English words both as calculus words to refer to real world entities and as stand-in words for the real world entities.

Example 0.6. (Continued) The French *word* "pomme" would then *stand-in* for, as well as *refer* to, the real world *entity*.

6 Coping With The Paper World

The meaning of all the calculus words to be employed in *RBC* cannot of course be **explained** in this Preface You *Don't* Have To Read but will appear progressively throughout *RBC* , when and as needed. The following is only about the *way* *RBC* deals with the semantics of the calculus language.

[27] https://www.researchgate.net/publication/346528673_A_Model_Theoretic_Introduction_To_Mathematics_4th_edition

1. Calculus words. In order for the meaning of calculus words to be precise, each and every calculus word will be explained with: **(a)** ordinary English words and calculus words that have *already* been explained, followed by **(b)** an **EXAMPLE** to illustrate how the calculus word is *used*.

define
meaningless

Most of the time, that will be enough for the reader to keep on trucking *safely* but, occasionally, it will be necessary to **define** a calculus word with a a **DEFINITION** that is a more formal explanation in terms of *only already* explained calculus words which will appear in a special format as in:

> **DEFINITION 0.1 Meaningless** is a synonym of "without meaning".

2. Diversity. But the chances of misunderstanding go beyond misunderstanding between *you*, the reader, and *RBC* or between *you* and *other* readers of *RBC* and there can also be misunderstanding between *you* and *other* texts and/or between *you* and readers of *other* texts, because:

> **LANGUAGE 0.2** The calculus language evolved as CALCULUS itself was being developed and different mathematicians, to help them focus on exactly what *they* were doing, often re-defined calculus words that already had a meaning given by other mathematicians with *their* definitions.
>
> In any case, other than the word pathological every once in a while, *RBC* will *never* employ words that *mathematicians* often employ but never really define![a]
>
> [a]`https://en.wikipedia.org/wiki/List_of_mathematical_jargon`)

3. Lightness. A danger for a text that wants to *explain* is to explain *too much* and thereby become insufferable. So, in order to lighten things up, *RBC* will not be above taking liberties with the calculus language but,

> **AGREEMENT 0.3** Any particular shortcut, such as abbreviating long words by letting parts going without saying and/or other such liberties will always be acknowledged by an **AGREEMENT** in this format.

symbol
notation
calculate
compute
iff

CAUTION 0.1 While unavoidable, letting words go without saying and depending on the context as a reminder is dangerous so to help accustom readers to parts of words eventually going without saying by the terms of an **AGREEMENT**, *RBC* will usually hint *for a while* at what will eventually go without saying.

EXAMPLE 0.6. (Continued) *For a while*, "pomme" might be clarified with some hint within parentheses such as:

(object language) "pomme"

or

(stand-in) "pomme"

4. Symbols. The calculus language does not consist only of calculus *words* but also includes **symbols** and **notations**[28] involving symbols.

i. While ordinary English does not lend itself to **calculations**—aka **computations**[29], the calculus language includes many symbols to allow for calculating.[30]

EXAMPLE 0.7. Figuring what would be left of three thousand seventy nine Dollars and eight Cents after spending six hundred forty seven Dollars and twenty six Cents would be a lot harder in *ordinary English* than *computing* the difference in the BASE TEN *language*[a]:

$$\begin{array}{r} \$3\,079.08 \\ -\$647.26 \\ \hline \end{array}$$

But then we could just spend *the money to see what's* left*!*

[a] https://en.wikipedia.org/wiki/Hindu%E2%80%93Arabic_numeral_system

However, the description of both symbols and notations that *RBC* will employ does not belong to this Preface You *Don't* Have To Read and will be described as needed.

ii. But not all symbols will be for computational purposes and a few are just like abbreviations. For instance, *RBC* will employ the following two symbols which, although standard, are relatively recent inventions with which the reader may not be acquainted:

[28] https://en.wikipedia.org/wiki/Mathematical_notation
[29] https://en.wikipedia.org/wiki/Calculation
[30] https://en.wikipedia.org/wiki/Mathematical_notation

Language 0.3 iff, read "if and only if", is the symbol that indicates of two sentences that neither sentence can be TRUE without the other sentence also being TRUE and therefore that neither sentence can be FALSE without the other sentence also being FALSE.[a]

[a] https://en.wikipedia.org/wiki/If_and_only_if

□

iff is not to be confused with IFF[31]

Example 0.8. The sentence "Jack is to the right of Jill iff Jill is to the left of Jack" is TRUE.

But why is "Jack sits to the right of Jill iff Jill sits to the left of Jack" FALSE?

Language 0.4 □, read "Q.E.D.", is the symbol that indicates the end of a proof.[a]

[a] https://www.urbandictionary.com/define.php?term=QED

5. No pronoun. And, last but not least, because it is extremely easy not to remember and/or not to see for which previous noun in a sentence a pronoun stands for, *RBC* tries never to use pronouns even at the cost of having to repeat the noun itself.

Pace English teachers!

Example 0.9. Instead of saying:

The mountain has a forest and a lake and it is beautiful.

RBC would say:

The mountain has a forest and a lake and
- the mountain
- the forest
- the lake
- the mountain with the forest
- the mountain with the lake
- the forest with the lake
- the mountain with the forest and the lake

(*whichever is intended*) is beautiful.

7 Reading *RBC*

To begin with, while reading Mathematics need not be forbidding, there is no denying that reading Mathematics is *never* easy.

No matter who you are: "Don't just read it; fight it! Ask your own questions, look for your own examples, discover your own proofs." *Paul R. Halmos,* I Want to be a Mathematician[32]

[31] https://en.wikipedia.org/wiki/Identification_friend_or_foe
[31] https://en.wikipedia.org/wiki/Paul_Halmos

reason given

1. Reading mathematics in general. The first thing to be emphasized is that it is impossible for *anybody* to get from a *single* reading of just about *any* part of *any* scientific text everything that's there.

This is because it is impossible for *any* piece of *any* scientific text to "say it *all*" because *any* piece of text will have to rely on some things having been said *earlier*, to prepare the ground, and some things can only be said *later*, when everything has been made ready to nail down the matter.

So, to begin with, the first thing people thinking of reading *RBC* ought to realize is that *nobody* can understand any scientific text, let alone mathematics text, not even *this* one, in just *one* reading. Absolutely nobody. For *RBC* really to make **reasonable** sense to *you*, *you* will have to re-read *RBC* , more than once.

In other words, you got to give reason *a reasonable time to* think *in. (Sorry, couldn't help it.)*

And, in particular, there are a couple of standard maneuvers used by *mathematicians* when they are reading a text and, like you will too, run into something they don't get:

- **Back & Forth maneuver!** If, even after you have made sure of the meaning of every single calculus word in the piece of text you are having trouble with, you know you still don't really get the message or something still does not make sense to you, then try going *back* to a place in the text with which you have made your peace and reread it anyway. You will probably discover things you had not thought of when reading it the first time. Now read forward till you reach that place where you stalled and it may very well be that those new things you hadn't thought of before will now help you make it through.
- **Wait & See maneuver.** If you *do* get what a piece of text is saying but just don't *really* see what the "point" is, make a *note* of your misgivings and keep on reading. Eventually you will probably have the "Aha", that is you will now realize that the "point" of the piece of text you had trouble with was to support what you are reading now.

Like you might finally really *see the reason for something in Relations Given By Data only somewhere in* **??**.

Finally, and more generally, even though, along with the discussions, there will always be **EXAMPLES** of what's being discussed, in order to *really* understand what is going on, *you*, the reader, willl have to **give** yourself other instances and examples of whatever is being discussed which is why the word give will appear very frequently as a reminder for *you* to give yourself, and discuss, your own **EXAMPLES**.

Remember, there will be no **EXERCISE** *in RBC and it will be entirely up to* you *to wonder about matters.*

In any case, though, the best approach is for two or three people to read the text *simultaneoulsy* but *separately* and then to confront their understandings.

Altogether then, and for whatever it is worth, the first way *RBC* claims to be different is the *explicit* attention being paid to matters of language.

Explicit? Extreme? Excessive?

2. The two major obstacles. Because you will want to think about what's *going on* rather than try to memorize what *RBC* is saying, *RBC* wants to be as *immediately* transparent as at all possible. However, there two obstacles

pdf click

- What goes without saying
- What is too "costly" to say it completely precisely. Not favorable to hard questioning

The fact that many calculus words are just ordinary English words to which a very precise meaning has been assigned is a major obstacle to learning the calculus language as the danger is for the reader facing later a calculus word to forget the precise meaning of the calculus word and to go by the meaning of the ordinary English word. Which, unfortunately, is exactly when things will stop making sense.

And therefore to understanding *CALCULUS,*

And, to make things even worse, *RBC* will have to use these calculus words alongside ordinary English word because it is of course with ordinary English words that *RBC* will describe and discuss *what* will be done with the calculus words and explain *why* things are being done that way.

Why do some people brandish mathematics *words like* space, catastrophe, field, category*, . . . whose meaning they don't really know? To try to impress you!*

And, achieving transparency isn't easy either.

3. Looking up the Index. Since, at least initially it is not easy to keep in mind precisely *what* calculus words refer to, like any standard book, *RBC* will help you retrieve what calculus words and symbols precisely refer to by having every single one of these calculus words and symbols listed in the INDEX at the end of the book along with the page where the calculus word or symbol appears in bold black characters in the text—as well as in red characters at the top of the margin of that page—and is explained and/or defined.

Onscreen, a click on the page number will get you there.

Using the INDEX more than occasionally, though, even onscreen, is a huge pain which makes it extremely likely you will put off looking up what the calculus word refers to precisely and rely instead on the ordinary English word, . . . and then be left facing text that makes no sense.

4. Clicking. And so, another thing that is different with *RBC* is that *RBC* was written to be read in **pdf**[33] form so that, *onscreen*, once introduced calculus word will always appear in reddish characters and **clicking** on that calculus word will instantly get you to the page where the word is explained.

In fact, and more generally,

As already mentioned on the title page. But what to click on to return to where you were will depend on your pdf *reader.*

[33] https://www.adobe.com/acrobat/about-adobe-pdf.html

factual evidence
general statement

AGREEMENT 0.4 Anything, anywhere, that appears in reddish characters is a click away from what that thing is about:

- *Titles* in all tables of contents,
- *Page numbers* in all references,
- *References* as in DEFINITION B.2 or **??** or as in the *Blue Note* just in the margin.

Want to take a break from this Preface You Don't *Have To Read? Just click on, for instance, To be or not to be functional or Compact views.*

However,

CAUTION 0.2 Calculus words are *not* clickable in either **EXAMPLES** or **DEMOS**, the idea being this will "incite" you to get back to whatever *explanation*, **DEFINITION** or **PROCEDURE** the **EXAMPLE** or the **DEMO** is an illustration of—and where calculus words *are* clickable.

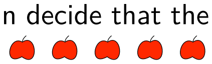
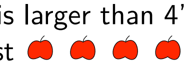
8 Proof Vs. Belief?

A much debated issue—at least by some *people.*

Another way *RBC* claims to be different has to do with the way *RBC* deals with the question of how to decide if a sentence is TRUE or FALSE or whether the truth of the sentence might be undecidable?

1. In everyday life. With some isolated statements, it is possible to decide whether of not the statement is TRUE or FALSE on the basis of **factual evidence**, that is by checking what the statement is saying directly against the real world.[34]

Unfortunately, the truth of most statements cannot be obtained by checking against the real world.

EXAMPLE 0.10. We can decide that the statement "4+1 is larger than 4" is TRUE by trying to match  one to one against . But can we do that with the statement "400 000 000 000 000 000 000+1 is larger than 400 000 000 000 000 000 000" ?

And **general statements** are simply impossible to check against the real world.

EXAMPLE 0.10. (Continued) And, even worse, what should we look at in the real world to decide if the statement "*Any* number plus one is larger than

[34] https://en.wikipedia.org/wiki/Verification_and_validation

the number itself" is TRUE?
Of course, we can check for any number(s) we want but we can't go on checking for ever and so we will never know for absolutely sure that the general statement "*Any* number plus one is larger than the number itself" is TRUE.

believe
assert
trust
theory
postulate
axiom
prove
theorem
natural deductive rule
Gödel
Gödel's Completeness Theorem

And, contrary to what many people seem to **believe** these days, just **asserting** a statement, no matter how many times and/or how forcefully, does not *make* that statement TRUE. ... And just invoking some other text doesn't work either: maybe the author of that other text had some hidden agenda? Or didn't really know what they were writing about? Or made some honest mistake?

On the other hand, even in everyday life, one cannot believe that each and every sentence being asserted is going to be TRUE.

EXAMPLE 0.11. What would happen, even in *everyday life* if, for instance, the result of an addition was up to the beliefs of whoever does the addition?

Of course, under threat of a gun *who wouldn't agree that, say,* $2\,000 = 7$*? But ...*

So, some *explicit* way to decide is necessary even if, at least in *everyday life*, the matter eventually comes down to being indeed a matter of belief—and therefore of **trust**.

2. In Mathematics. Scientists and mathematicians on the other hand are not interested in the truth or falsehood of isolated statements but in the description of the real world with **theories**[35] that is collections of sentences obtained as follows:

To see very simple examples of how this works, you may want to download the author's A Model Theoretic Introduction To Mathematics[39]

i. Postulate[36] those sentences that will be the **axioms**[37] of the theory, that is list the few sentences *believed* to have to be in the theory,
and then

ii. Prove that other sentences are also **theorems**, that is are also in the theory, by employing **natural deductive rules**[38] on axioms and/or sentences already proven to be theorems.

Then, because of **Gödel's Completeness Theorem**[40], the truth of a sentence will derive from the truth of the axioms and of the sentence(s) from which the natural deductive rules proved that the sentence was a theorem.

[35] https://en.wikipedia.org/wiki/Theory#Mathematical
[36] https://www.thefreedictionary.com/postulate
[37] https://en.wikipedia.org/wiki/Axiom
[38] https://en.wikipedia.org/wiki/Natural_deduction
[39] https://www.researchgate.net/publication/346528673_A_Model_Theoretic_Introduction_To_Mathematics_4th_edition
[40] https://en.wikipedia.org/wiki/G%C3%B6del%27s_completeness_theorem

Thus the natural deductive rules ultimately reduce the question TRUE or FALSE about each one of the many theorems in the theory to the question TRUE or FALSE about only the few axioms underlying the theory. (Which is *not* to say that all sentences in the (object) language can be proven to be true or false.[41])

Sometimes, though, axioms are picked just out of curiosity, just to see what theorems could be proven from postulating these axioms as opposed to those other axioms. No end to curiosity.

Proofs are checked by making them available to the relevant part of the mathematical community and there is thus a kind of "social contract".[42]

On what *basis* the axioms are chosen, though, is a totally separate issue. Axioms are sometimes *conjectured*[43] on the basis of some observation of the real world but usually on the basis of some already "accepted" theory. So, since belief is based on trust, when all is said and done, which axioms you choose is a matter of trust, much like what "the rest of us" do in the real world.

Hopefully, though, the proofs will be good enough to be "convincing arguments".

> **CAUTION 0.3** Since readers of *RBC* have yet to become mathematicians, the way **THEOREMS** will be proven in *RBC* will bear only a distant resemblance with proofs as understood by mathematicians and described above.

3. How falsehood can spread even in mathematics. We must always keep in mind, though, that deductive rules can spread falsehood like wildfire.

EXAMPLE 0.12. One of the deductive rules in ALGEBRA is that "*adding equals to equals yields equals*". Now:

- If we accept a TRUE sentence like $4+5$ and $6+3$ are equal as a theorem, then adding, for instance, 7 to each of $4+5$ and $6+3$ will force us to accept the sentence $4+5$ $+7$ and $6+3$ $+7$ are equal as a theorem too which is fine inasmuch as the sentence 16 and 16 are equal is indeed TRUE,

But:

- If we accept a FALSE sentence like $4+5$ and $6+2$ are equal as a theorem, then:
  - adding, for instance, 7 to each of $4+5$ and $6+2$ will force us to accept

[41] https://en.wikipedia.org/wiki/G%C3%B6del%27s_incompleteness_theorem

[42] https://www.quantamagazine.org/why-mathematical-proof-is-a-social-compact-20230831/?mc_cid=0ade39707d

[43] https://en.wikipedia.org/wiki/Conjecture

rigor
Laurent Schwartz

the sentence $4+5+7$ and $6+2+7$ are equal as a theorem too which is unfortunate inasmuch as the sentence 16 and 15 are equal is in fact FALSE. And then, even worse,

– adding, for instance, 7 to each of $4+5+7$ and $6+2+7$ will now force us to accept the sentence $4+5+7+7$ and $6+2+7+7$ are equal also as a theoren which is unfortunate inasmuch as the sentence 23 and 22 are equal is in fact FALSE. And then, still worse,
– adding, for instance, 7 to each of $4+5+7+7$ and $6+2+7+7$ will now force us to accept the sentence $4+5+7+7+7$ and $6+2+7+7+7$ are equal also as a theorem which is unfortunate inasmuch as the sentence 30 and 29 are equal is in fact FALSE. And then, even still worse,
– Etc

4. In the sciences. Just to clarify,

CAUTION 0.4 A *scientific* theory[a] is a much more complicated thing than a *mathematical* theory.

[a]https://en.wikipedia.org/wiki/Theory#Scientific

But you don't have to worry since, after all, RBC is in MATHEMATICS.

9 Reason Vs. Rigor

So, since the foremost fear in MATHEMATICS is making a mistake in a proof, and thereby getting as theorem a sentence which may actually be FALSE, *mathematicians* proceed as **rigorously**[44] as possible, that is provide as many steps in the proof as they possibly can—that is while remaining "*readable*"—*and* are always able, willing and ready to provide missing steps on demand.

That's what refereeing is all about.

Unfortunately, CALCULUS has been extraordinarily difficult to develop *rigorously*

EXAMPLE 0.13. While 'Delta functions'[a] had been used since the early eighteen hundreds, it was only in 1950 that **Laurent Schwartz**[b] was awarded the Fields Medal for having defined 'Delta functions' rigorously.

[a]https://en.wikipedia.org/wiki/Dirac_delta_function

[44]https://en.wikipedia.org/wiki/Rigour#Mathematical_rigour

Whatever 'Delta functions' are: the single quotes, ' ', say that, at this time, you are neither assumed nor supposed to know what 'Delta functions' are. analysis'

[b]https://en.wikipedia.org/wiki/Laurent_Schwartz

As a result, the number one question for authors of CALCULUS texts is how rigorous to be. A few texts, titled REAL ANALYSIS [45], are as completely rigorous as at all possible but the rest, just titled CALCULUS, are *far* from being rigorous as they skip whatever the author thinks will be too much for the buyers.

But, while *RBC* is just as far as standard texts from being rigorous, there is a very big difference: standard texts retain, however un-rigorously, the *modus operandi*[46] of *mathematicians* while *RBC* aims at communicating the way *hard scientists*[47] and *engineers* have long understood and used CALCULUS—without worrying one bit about its lack of rigor.

As physicist David Hestenes[48] *said at the outset of his 2002 Oersted lecture:*
Taking course content as given [...] ignores the possibility of improving pedagogy by reconstructing course content.

Lastly, the conformist reader ought to be reminded that, instead of being based on limits or infinitesimals, as it seems *all* current CALCULUS textbooks are:

CAUTION 0.5 In *RBC*, CALCULUS will be developed by way of polynomial approximations which are the equivalent in CALCULUS of the decimal approximations used by *scientists* and *engineers* in **applications** of CALCULUS to the real world.

And here, Ladies and Gentlemen, is where this Preface You Don't *Have To Read finally comes to an end.*

[45]https://en.wikipedia.org/wiki/Real_analysis
[46]https://en.wikipedia.org/wiki/Modus_operandi
[47]https://en.wikipedia.org/wiki/Hard_and_soft_science

When you have mastered numbers, you will in fact no longer be reading numbers, any more than you read words when reading books. You will be reading meanings.[14]

Harold Geneen [15]

Chapter 0

Reasonable Numbers

The very purpose of *RBC*, namely to help people "*develop* a CALCULUS they can *use* in the real world" (For Whom *This* Text?, Page xiii), makes it necessary to begin with two separate questions about numbers that, unfortunately, are seldom dealt with in ARITHMETIC texts:

- What kind of numbers are needed to develop such a CALCULUS?
- What kind of numbers are needed to use such a CALCULUS?

Before anything else, though, an important instance of our Use of ordinary English words (AGREEMENT 0.2, Page xx) will be that:

No, no, this is not *going to be your standard* Review Of Basic Skills You Shouldn't Have Forgotten!
And you really should *read this Chapter if only just to have an* idea *of what's in it.*
And don't panic: as you go on, you will always *be able to click anything you have trouble with.*
And, eventually, it will all make perfect sense*.*

AGREEMENT 0.1 In this Chapter 0, the ordinary English word "number" (https://en.wikipedia.org/wiki/Number) will be used both as a *stand-in word* for, and as a ordinary English word to *designate* the various real world entities which are *designated* in the ARITHMETIC language by the word **numeral** (https://en.wikipedia.org/wiki/Numeral).

[14] As discussed very thoroughly in https://history.stackexchange.com/questions/45470/source-of-quote-attributed-to-w-e-b-du-bois-when-you-have-mastered-numbers?rq=1, this famous quote is *not* from W.E.B. Dubois, as often asserted—with no reference, but from page 151 of *Managing*, Chapter Nine - The Numbers, Geneen's book.

[15] (https://en.wikipedia.org/wiki/Harold_Geneen.)

number phrase
denominator
quality
numerator
quantity
discrete aspect
collection
item
collection of items
Cantor
set
element

We will *not use* the word numeral and introduce calculus words to designate these various real world entities which are *designated* in the ARITHMETIC language by the word **numeral**

1 Numbers In The Real World

To begin with, in the real world, numbers are not used all by themselves as in ARITHMETIC textbooks but in **number phrases** consisting of:

- A **denominator** which is a noun indicating **quality** by saying *what* is being numbered,

together with

- A **numerator** which is a number indicating **quantity** by saying *what* the numbering resulted in.

EXAMPLE 0.1. The following might occur in the real world:
3 **Apples**, 5 **Feet**, 72.4 **ºFarenheit**,
$\frac{3}{\mathbf{8}}$ **Inch**, where 3 is tne numerator and "of which **8** make up an **inch**" is the denominator.

And since there are many different aspects to the real world there are many different kinds of number phrases and different kinds of numbers.

But fundamentally there are two basic aspects to the real world that need to be discussed briefly.

1. Discrete aspect of the real world. The simpler aspect of the real world is the **discrete aspect** which involves **collections of items** where, to quote **Georg Cantor** [16](1845 - 1918), the creator of SET THEORY[17]: "*By an "aggregate" (Menge) we are to understand any collection into a whole (Zusammendfassiung zu einem Ganzen) M of definite and separate objects* m *of our intuition or our thought. These objects are called the "elements" of M.*"[18]

EXAMPLE 0.2. The *discrete* aspect of light is as a collection of *photons*.[a]

[a] https://en.wikipedia.org/wiki/Photon

[16] https://en.wikipedia.org/wiki/Georg_Cantor
[17] https://en.wikipedia.org/wiki/Set_theory
[18] https://en.wikipedia.org/wiki/Naive_set_theory

LANGUAGE 0.1 *RBC* does *not employ* the words **set** and **element**, now standard in mathematics [a], because the ordinary English words collection and item are immediately transparent while the words set and element might seem to call for a knowledge of SET THEORY which is completely irrelevant for RBC

[a] Actually, the original English translations for Cantor's word "Menge" *was* the word collection. As for the word element, see for instance https://english.stackexchange.com/questions/85648/what-are-the-differences-between-element-and-item-regarding-a-list

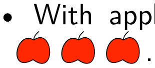
whole number
information
plain whole number
signed whole number
list

When dealing with collections of items,

- The denominator in the number phrase for a collection of items simply denotes the kind of items in the collection.
- The numerator in the number phrase for a collection of items will be a **whole number** but the *kind* of whole number will depend on the kind of **information** that is wanted about the collection of items:
  - ***Plain* whole numbers** when all the information that is wanted is *how many* items there are in the collection of items
  - ***Signed* whole numbers** when the information that is wanted is both *how many* items there are in the collection of items and *which way* the collection of items is going

Even the fabulous Bourbaki's Theory Of Sets starts: "From a "naive" point of view, many mathematical entities can be considered as collections or "sets" of objects. *"!*

EXAMPLE 0.3. When dealing:

- With apples, 3 **Apples** is the number phrase that denotes the collection .
The numerator is the *plain* whole number 3 and the denominator is **Apples**.
- When dealing with boxes of bananas, -5 **Boxes of bananas** is the number phrase which denotes a collection of five boxes of bananas on its way out, or being owed, or etc.
The numerator is the *signed* whole number -5 and **Boxes of bananas** is the denominator.

On Equal Exchange of course. (https://en.wikipedia.org/wiki/Equal_Exchange.)

Collections of items can be **listed** and lists are very useful to organize information[19].

More generally, DISCRETE MATHEMATICS is the part of mathematics dealing with the discrete aspect of the real world[20].

[19] https://en.wikipedia.org/wiki/Information
[20] https://en.wikipedia.org/wiki/Discrete_mathematics

continuous aspect
amount of stuff
amount
stuff
unit of stuff
decimal number
plain decimal number
signed decimal number
decimal point

2. Continuous aspect of the real world. The other, more complicated, aspect of the real world is the **continuous aspect** which involves **amounts of stuff**. (https://en.wikipedia.org/wiki/Continuum_(measurement).).

EXAMPLE 0.4. The *continuous* aspect of light is as an amount of *radiation*. (https://en.wikipedia.org/wiki/Light.)

When dealing with amounts of stuff,

- The denominator in the number phrase requires the prior definition of a **unit of stuff** which is then used as denominator.
- The numerator in the number phrases for an amount of stuff will be a **decimal numbers** but the *kind* of decimal number will again depend on the kind of information that is wanted about the amount of stuff:
  - ▸ ***Plain* decimal numbers** when all the information that is wanted is *how much* stuff there is in the amount of stuff
  - ▸ ***Signed* decimal numbers** when the information that is wanted is both *how much* stuff there is in the amount of stuff and *which way* the amount of stuff is going

EXAMPLE 0.5. When dealing with **milk**, after we have taken  as unit of **milk**,

- 3.4 **Gallon of milk** is the number phrase that denotes the amount of **milk** in .

The numerator is the *plain* decimal number 3.4 and the denominator is the unit of **milk**, namely **Gallon of milk**

- $+5.7$ **Gallon of milk** is the number phrase which denotes an amount of milk on its way in, or being due in, or etc.

The numerator is the *signed* decimal number $+5.7$ and the denominator is the unit of **milk**, namely **Gallon of milk**.

AGREEMENT 0.2 In RBC, decimal numbers will *always* be written with a **decimal point** to the *right* of some digit.

EXAMPLE 0.6.

- ▸ $.783$ is *not* a plain decimal number because the decimal point is *not* to the right of a digit,
- ▸ $-783.$ is a signed *decimal* number,

▸ 0.27 is a plain decimal number.

signed number
rational number
irrational number
fraction
mixed number
real number
complex number
count
measure
error

And, finally, a matter to be discussed in Giving Numbers (Section 3, Page 8),

> **Caution 0.1** **Signed numbers**, whether (signed decimal) numbers or signed whole numbers, do *not* include zero which will be discussed in Open numbers vs. fixed numbers

Since Calculus is the part of mathematics dealing with the continuous aspect of the real world (https://en.wikipedia.org/wiki/Calculus), *RBC* will employ signed decimal numbers and plain decimal numbers but:

> **Caution 0.2** *RBC* will *never* employ **real numbers** and or any other numbers such as **fractions**, **mixed numbers**, **rational numbers**, **irrational numbers**, **complex numbers**, etc .

If you want to know why no *so-called real numbers, see* ??.

3. Whole numbers vs. decimal numbers. Even though Calculus deals with amounts of stuff, *RBC* will often *use* collections of items, and therefore whole numbers in **Examples** and **Demos** .

Moreover, it will occasionally be enlightening to contrast some aspects of whole numbers with the corresponding aspects of decimal numbers. For instance:

- We get the plain whole number which is the size of a collection of items by **counting** the items in the collection (https://en.wikipedia.org/wiki/Counting),

but

- We get the plain decimal number which is the size of an amount of stuff by **measuring** the amount of stuff, (https://en.wikipedia.org/wiki/Measurement).

This is a *major difference* because

- Counting a collection of items is, at least usually, simple enough, and provides a single, definite plain whole number,

while

- Measuring an amount of stuff is complicated because, not only does measuring *unavoidably* involve making some **error** due to such matters as the quality of the equipment and/or the ability of the user of the equipment,

uncertainty
measured number
Timothy Gowers
digit
non-zero digit
leading zero
trailing zero

but also because there is *unavoidably* going to be some **uncertainty** in the size of the error and therefore in the **measured number**.

EXAMPLE 0.7. We cannot really say "I drank 2.3 **cups of milk**" because how much milk we really drank depends on the care with which the amount of milk was measured, how much was left in the bottle, etc. The *uncertainty* may of course be too small to matter . . . but then may not.

In fact,

> As **Timothy Gowers** (https://en.wikipedia.org/wiki/Timothy_Gowers, (Fields Medal 1998) puts it in the 6th paragraph of https://www.dpmms.cam.ac.uk/~wtg10/continuity.html):
> "*Physical measurements are not real numbers. That is, a measurement of a physical quantity will not be an exactly accurate infinite decimal. Rather, it will usually be given in the form of a finite decimal together with some error* ***estimate****:* $x = 3.14 \pm 0.02$[a] *or something like that.*"
>
> [a]See EXAMPLE 0.16 (Page 10)

Right now, *you won't be able to make much sense of the rest of Gowers' text but even a cursory glance will show his concern with the real world.*

2 Issues With Decimal Numbers

This is a majpr issueswith signed decimal numbers which, while not *directly* relevant to the *development* of CALCULUS, is most important for the *use* of CALCULUS in the *experimental sciences* and *engineering*.

1. How many digits in a number. To begin with, it is not immediately obvious how many **digits** (https://en.wikipedia.org/wiki/Numerical_digit) there really are in a signed decimal number because one of the digits being used to write signed decimal numbers, namely the *digit* 0, behaves very differently from the **non-zero digits** if only because of **leading zeros** (https://en.wikipedia.org/wiki/Leading_zero) and **trailing zeros** (https://en.wikipedia.org/wiki/Trailing_zero).

EXAMPLE 0.8. How many digits are there in 00000000.00120034000000 ?
Answer: Nine, namely 0.00120034
Because, if we take out any more 0, we will be left with either a different decimal number, say 0.0120034 or 0.0012034, and, if we don't leave at least

one 0 *before* the decimal point, we will be left with .00120034 which is *not* a decimal number because there is no digit being *pointed*.

figure
significant digit
Of course, it's being done all the time but it won't in RBC.

LANGUAGE 0.2 **Figure** is a word often used instead of digit but *RBC* will stick with the word digit,

2. Importance of the digits. Not all digits in a decimal number have the same importance.

a. The more non-zero digits a signed decimal number has, whether left or right or both sides of the decimal point, the less likely the signed decimal number is to designate anything in the real world.

EXAMPLE 0.9. None of the following numbers is likely to designate anything in the real world:

+602 528 403 339 145 485 295 666 038 294 891 392 775 987 378 000 261 234 386 384 558 003 000 384 203 771 790 349 303 000 000 000 809 234 329 262 234 085 108 500 000 002 891 038 456 666•

−0•602 528 403 339 145 485 295 666 038 294 891 392 775 987 378 000 261 234 386 384 558 003 000 384 203 771 790 349 303 000 000 000 809 234 329 262 234 085 108 500 000 002 891 038 456 666

+602 528 403 339 145 485 295 666 038 294 891 392 775 987 378 000 261 234 386 384 558 003 000 384 203 771 790 349 303 000 000 000 809 234 329•262 234 085 108 500 000 002 891 038 456 666

b. Indeed, only so many digits in a signed decimal number can be **significant digits**, that is can correspond to any particular precision in the measurement. (https://en.wikipedia.org/wiki/Significant_figures

EXAMPLE 0.10. To say that "*the estimated population of the US was "*328 285 992 *as of January 12, 2019*" (https://en.wikipedia.org/wiki/DEMOgraphy_of_the_United_States on 2019/02/06) is not reasonable because at least the rightmost digit, 2, is certainly *not* a significant digit: on that day, some people died and some babies were born so the population could just as well have been designated as, say, 328 285 991 or 328 285 994.

The numbers in https://en.wikipedia.org/wiki/Iron_and_steel_industry_in_the_United_States are much more reasonable: '*In 2014, the United States [...] produced' 29 million metric tons of pig*

reasonable signed decimal number
open number

iron and 88 million tons of steel." Similarly, "*Employment as of 2014 was 149,000 people employed in iron and steel mills, and 69,000 in foundries. The value of iron and steel produced in 2014 was 113 billion.*"

3. Issues with significant digits.. There are two main issues:

- Deciding which digit(s) in the measurement of an amount of stuff, if any, is/are significant digits which depends on the precision with which the real world is being measured and, as such, is not directly relevant to CALCULUS.
- Deciding how many digits in the result of a calculation are significant digits (https://en.wikipedia.org/wiki/Significant_figures#Arithmetic) which is essentially a matter of ARITHMETIC rather than of CALCULUS .

However, neither one of these issues will be of concern in *RBC* and **reasonable signed decimal numbers** will be signed decimal numbers with only so many significant digits.

As well as people "just" playing with CALCULUS.
Isn't that fortunate!

3 Giving Numbers

When all has been said and done, in *RBC* CALCULUS is about being *used* by non-mathematicians and for non-mathematical purposes and this raises paricular issues that, at the very least, need to be acknowledged.

The fact that Calculus eventually has to used by non-mathematicians and for non-mathematical purposes is often overlooked and a few issues will be discussed here.

That is, in the instance, by you, the reader.

the following issues will have to distinguished:

- Issues users will face *after* a number has been given,

from

- Issues users will face *before* a number has been given.

1. Open numbers vs. fixed numbers. https://en.wikipedia.org/wiki/Mathematical_constant

https://en.wikipedia.org/wiki/Constant_(mathematics)

A difficulty the user faces even before giving a number is determining whether, in a situation, a number is:

- ► An **open number** in the situtation, that is a number that *can* be changed to any other number in that situation.

EXAMPLE 0.11. The people of Jacksville are considering paving part of the parking lot next to Township Hall and since both the *length* and the

fixed number
given number
generic given number
x_0
x_1
x_2
y_0
y_1
y_2

width of the area to be paved are open numbers, people are discussing the pro and con of 20ft long by 100 feet wide versus 60ft long by 60 feet wide versus 100ft long by 30 feet wide versus etc, etc.

or

- A **fixed number** in the situtation, that is a number that *cannot* be changed to any other number in that situation.

EXAMPLE 0.12. The people in Jillsburg are considering paving part of the road from City Hall down to the river but, since the *width* of a road is *fixed* by the County, only the *length* of the area to be paved is an *open number* and people are discussing the pro and con of 300 ft long versus 1000 ft long versus 500 ft long versus etc, etc.

> **LANGUAGE 0.3** What we call fixed numbers are also called **constants**.

EXAMPLE 0.13. The circumference of a circle of diameter 702.36 is equal to 3.14159×702.36
(https://en.wikipedia.org/wiki/Circumference#Circle),
Here, 702.36 is an *open* number because 702.36 can be replaced by any other number to get the circumference of a circle with that diameter but 3.14159 is a *fixed number* that *cannot* be replaced by any other number. (https://en.wikipedia.org/wiki/Pi)

2. Generic given numbers. CALCULUS is *used* with **given numbers**, that is with numbers to be given by the *user*.

But the difficulty in explaining issues that will face the user *after* a number has been given is that *RBC* don't know *what* number the user will have given.

So, while the user will indeed be the one to give numbers, at least in the **PROCEDURES**, *RBC* , will have to employ ***generic* given numbers**, that is temporary *substitutes* for the numbers eventually to be given by the *user*:

> **DEFINITION 0.1** The generic given numbers will be the symbols: $\boldsymbol{x_0}$, $\boldsymbol{x_1}$, $\boldsymbol{x_2}$, etc, and $\boldsymbol{y_0}$, $\boldsymbol{y_1}$, $\boldsymbol{y_2}$, etc.

specify
required number
tolerance
cap

But of course, in **EXAMPLES** and **DEMOS** *RBC* will just give *actual* given numbers.

3. Specifying an amount of stuff. Because of the uncertainty intrinsic to measurements, there is more to **specifying** an amount of stuff (https://en.wikipedia.org/wiki/Specification_(technical_standard)) than just giving the **required number**:

Actually, this goes for large *whole numbers too.*

CAUTION 0.3 A number cannot specify an amount just by *itself*:

So, *scientists* and *engineers* *use* specifications that consist of *two* numbers:

- A required number to designate the amount of stuff they *require*,
- A **tolerance**, that is a **cap** on the size of the error, that is on how much the measured number will be allowed to differ from the required number (https://en.wikipedia.org/wiki/Engineering_tolerance).

EXAMPLE 0.14. When we want to buy a amount of milk, say "6.4 quarts of milk", to find out if we got our money's worth, we will have to measure how much mild we got in return for our money and since measuring amounts of stuff involves an incertitude about the size of the error and so, in our specification, we have to put a cap on the size of the error we are willing to tolerate, say "0.02 quarts of milk".

EXAMPLE 0.15. We cannot specify a distance in light years with a tolerance in inches.

But, if rather unfortunately,

LANGUAGE 0.4 It is completely standard to write, as Gowers is quoted doing in Subsection 1.3 - Whole numbers vs. decimal numbers (Page 5)

$$x = x_0 \pm T$$

that is that the measured number is *equal* to the required number ± the tolerance which, strictly speaking, makes no sense!

EXAMPLE 0.16. Strictly speaking, it makes no sense to specify $+3.14 \pm 0.02$ because that would specify $+3.14 \oplus +0.02$, that is $+3.16$, or $+3.14 \oplus -0.02$, that is $+3.12$.

What is being specified by $+3.14 \pm 0.02$ is a required $+3.14$ with an error less than the tolerance 0.02, in other words any number *between* $+3.12$ and $+3.16$

variable place

CAUTION 0.3 - (Page 10) can then be restated in a more constructive manner:

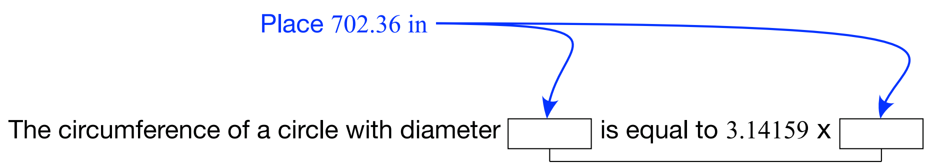
> **CAUTION 0.3 (Restated)** A required number together with a *tolerance* will specify an amount of stuff.

Of course, not all errors have the same relevance to the real world situation.

EXAMPLE 0.17. An error of less than $5.00 is devastating if the required number is $13.27 but complêtely insignificant if the required number is $1 018 000 008..
In other words, while the difference between $8.27 and $18.27 is the same as the difference between $1 017 999 995. and $1 018 000 005., namely $10., a tolerance of $10. is devastating if the required number is $13.27 but complêtely insignificant if the required number is $1 018 000 000..

4. Variables. In order to deal with issues *before* a number has been given, *RBC* could of course just leave the space empty to be eventually filled by the user with their given number.

EXAMPLE 0.18. The sentence at the beginning of EXAMPLE 0.16 (Page 10) could have been obtained from:



Instead of an empty space, though, *RBC* will employ the standard way which is to temporarily occupy the space with a **variable**, that is with a symbol that does *not* denote any particular number but acts as a **placeholder** eventually to be *replaced* by the user with a given number.
(https://en.wikipedia.org/wiki/Variable_(mathematics))

Or by RBC *with a generic given number.*

EXAMPLE 0.18. (Continued) Instead, *RBC* would employ a variable, say x, and write

global variable
x
y
z
semi-global variable
x_{pos}
y_{pos}
z_{pos}
x_{neg}
y_{neg}
z_{neg}
global expression

Replace x by 702.36 in
The circumference of a circle with diameter x is equal to $3.14159 \times x$

Er, so the meaning of variable is variable.

LANGUAGE 0.5 The calculus word variable is a *noun* but the ordinary English word *variable* is a adjective saying that something can *vary* and therefore entails the existence of *various* possibilities.

EXAMPLE 0.19. When the Weather Forecast is that tomorrow's weather is going to be "variable", they mean that the weather is not going to remain the same throughout the day.

Place-holder would of course be much more intuitive than variable but variable is historically entranched and absolutely untouchable.

Because numbers can occur in different ways, *RBC* will employ different *kinds* of variables which will be introduced later, as needed.

The first kind of variable *RBC* will employ will be:

In other words, while x_0 is a number that was given behind our back, x is a number that has yet to be given.

DEFINITION 0.2 The **global variables** are the place-holding symbols $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{z}$ which can be replaced by *any* given number.

We will also occasionally use

DEFINITION 0.3 The **semi-global variables** are the place-holding symbols

- $\boldsymbol{x_{\text{pos}}}$ $\boldsymbol{y_{\text{pos}}}$, $\boldsymbol{z_{\text{pos}}}$ which can be replaced by *any* given *positive* number,
- $\boldsymbol{x_{\text{neg}}}$, $\boldsymbol{y_{\text{neg}}}$, $\boldsymbol{z_{\text{neg}}}$ which can be replaced by *any* given *negative* number.

4 Expressions And Values

Just like ARITHMETIC, the very heart of CALCULUS will involve creating other numbers from given numbers but the way the other numbers will be created is more complicated.

1. Global expressions. A **global expressions in terms of a** global variable is a (grammatically correct) assemblages of symbols *with* at least one occurence of a global variable but *without* any calculus verb such as $=$, $<$, $\geqq$, . . .

individual expression
declare
execute

For a reason that will appear in Relations Given By Data-sets, *RBC* will show global expressions against a green background with the variable against a pink background.

EXAMPLE 0.20.

- $-17.03 \odot x$ is a global expression in terms of x ,
- $+2.73 \ominus -58.82$ is *not* a global expression because there is no variable,
- $-0.0021 \oplus y \otimes -5.01$ is a global expression in terms of y ,
- $\dfrac{x^{+2} \ominus +7}{x \oplus +3}$ is a global expression in terms of x ,
- $-0.0021 \oplus y < -5.01$ is *not* a global expression because of the verb $<$.

CAUTION 0.4 There are many different definitions of what a global expression is depending on the branch of mathematics and/or on the author's focus[a]

[a] https://en.wikipedia.org/wiki/Expression_(mathematics)

2. Individual expressions. Of course, what the *user* will want is to get an **individual expression** for a given number which is done by replacing the global variable in the global expression by the given number.

EXAMPLE 0.20. (Continued)

- $-17.03 \odot -73.042$ is an individual expression for -73.042 ,
- $+2.73 \ominus -58.82$ is *not* an individual expression because there was no variable,
- $-0.0021 \oplus y_2 \otimes -5.01$ is an individual expression in terms of y_2 ,
- $\dfrac{-5.008^{+2} \ominus +7}{-5.008 \oplus +3}$ is an individual expression in terms of -5.008 ,
- $-0.0021 \oplus +172.444 < -5.01$ is *not* an individual expression because of the verb $<$.

In standard CALCULUS texts, this is done in one quick single step but to keep things clear *RBC* will use:

PROCEDURE 0.1 To get from a global expression in terms of a global variable an individual expression in terms of a given number,

a. Declare the given number by writing to the right of the

global expression in terms of global variable

the declaration

| global variable to be replaced by given number

Altogether, then, we have:

global expression in terms of global variable | global variable to be replaced by given number

b. Execute the declaration, that is replace in the global expression every occurence of the global variable by the given number. We thus get the individual expression for the given number:

individual expression in terms of the given number

DEMO 0.1 Get from $\frac{x^{+2} \ominus +7}{x \oplus +3}$ the individual expression for $+5$

a. We declare $+5$ by writing to the right of

$$\frac{x^{+2} \ominus +7}{x \oplus +3}$$

the declaration

$$\Big|_{x \leftarrow +5}$$

Altogether, then, we have:

$$\frac{x^{+2} \ominus +7}{x \oplus +3}\Bigg|_{x \leftarrow +5}$$

b. We execute the declaration by replacing every occurence of the global variable x in the global expression by the given number $+5$. We thus get the individual expression for the given number $+5$:

$$\frac{{+5}^{+2} \ominus +7}{+5 \oplus +3}$$

generic individual expression
evaluate
AT
numerical value

Keep in mind, though, that other than in **Examples** and **Demos**, *RBC* will have to employ *generic* given numbers and that *RBC* will thus get ***generic* individual expressions**.

Example 0.20. (Continued) With the generic given number y_2

- $-0.0021 \oplus y_2 \otimes -5.01$ is a generic individual expression in terms of y_2

,

Agreement 0.3 The adjective generic in generic individual statement will go without saying.

3. Evaluation *AT* a given number. To **evaluate** a global expression ***AT*** a given number, that is to get the **numerical value** of the *individual* expression for the given number, *RBC* will employ:

Procedure 0.2 To evaluate a global expression in terms of x *AT* a given number x_0:

The first two steps are to Get an individual expression from a global expression while the third step is computational:

a. Declare the given number x_0 by writing the declaration

$\Big|_{x \leftarrow x_0}$, read "x *to be* replaced by x_0",

to the right of the global expression:

$$\text{global expression in terms of } x \Big|_{x \leftarrow x_0}$$

b. Execute the declaration by replacing every occurence of the global variable x in the global expression by the given number x_0 to get the individual expression for the given number x_0:

c. Try to carry out the computations in the individual expression in terms of x_0

DEMO 0.2a Evaluate $\dfrac{x^{+2} \ominus +7}{x \oplus +3}$ *AT* $+5$

a. We declare the given number $+5$ by writing the declaration

$$\Big|_{x \leftarrow +5}$$

, read "x *to be* replaced by $+5$",

to the right of the global expression:

$$\dfrac{x^{+2} \ominus +7}{x \oplus +3}\Bigg|_{x \leftarrow +5}$$

b. We execute the declaration by replacing every occurence of the global variable x in the global expression by the given number $+5$ to get the individual expression for the given number $+5$:

$$\dfrac{+5^{+2} \ominus +7}{+5 \oplus +3}$$

c. We try to carry out the computations in $\dfrac{+5^{+2} \ominus +7}{+5 \oplus +3}$:

$$\dfrac{+25 \ominus +7}{+5 \oplus +3}$$

$$\dfrac{+18}{+8}$$

$$+4.5$$

So, the numerical value of $\dfrac{x^{+2} \ominus +7}{x \oplus +3}$ *AT* $+5$ is $+4.5$

Unfortunatley, a global expression cannot necessarily always be evaluated *At* a given number because the computations cannot necessarily always be completed.

DEMO 0.2b Evaluate the global expression $\dfrac{x^{+2} \ominus +7}{x \oplus +3}$ *at* -3

a. We declare the given number -3 by writing the declaration $\Big|_{x \leftarrow -3}$, read "x *to be* replaced by -3", to the right of the global expression:

$$\left.\frac{x^{+2} \ominus +7}{x \oplus +3}\right|_{x \leftarrow -3}$$

b. We execute the declaration by replacing every occurence of the global variable x in the global expression by the given number -3 to get the individual expression for the given number -3:

$$\frac{-3^{+2} \ominus +7}{-3 \oplus +3}$$

c. We try to carry out the computations in

$$\frac{-3^{+2} \ominus +7}{-3 \oplus +3}$$
$$\frac{+9 \ominus +7}{0}$$
$$\frac{+2}{0}$$

but the computation comes to a halt before we get a numerical value because we cannot carry out the division.

And matters can easily turn out even more complicated. For instance:

Demo 0.2c Evaluate $\dfrac{x^{+2} \ominus +9}{x \oplus -3}$ *at* $+3$

a. We declare the given number $+3$ by writing the declaration $\Big|_{x \leftarrow +3}$, read "x *to be* replaced by $+3$", to the right of the global expression:

$$\left.\frac{x^{+2} \ominus +9}{x \oplus -3}\right|_{x \leftarrow +3}$$

b. We execute the declaration by replacing every occurence of the global variable x in the global expression by the given number $+3$ to get the individual expression for the given number $+3$:

$$\frac{{+3}^{+2} \ominus +9}{+3 \oplus -3}$$

c. We try to carry out the computations in $\frac{{+3}^{+2} \ominus +9}{+3 \oplus -3}$

$$\frac{+9 \ominus +9}{0}$$

$$\frac{0}{0}$$

but the computations come to a halt before yielding a numerical value because we cannot carry out the division.

5 Formulas And Sentences

In keeping with "*the first way RBC claims to be different is the* explicit *attention being paid to matters of language*"

1. Formulas. One thing that distinguishes the calculus language from ordinary English is that, as "*one of the most influential figures of computing science's founding generation, Edsger Dijkstra,*" once said, "*A picture may be worth a thousand words,* [but] *a formula is worth a thousand pictures.*[21] and so:

i. While *RBC* will deal only later with equations and inequations—which are essentially formulas,

ii. The calculus word "sentence" needs to be introduced in a manner consistent with what precedes because explanatory texts are often circular.

Example 0.21. "*In mathematics, a formula generally refers to an equation relating one mathematical expression to another*" [a] but then "*an equation is a mathematical formula that expresses the equality of two expressions, by connecting them with the equals sign* $=$"[b]

[21] https://en.wikipedia.org/wiki/Formula#In_computing

[a]https://en.wikipedia.org/wiki/Formula#In_computing
[b]https://en.wikipedia.org/wiki/Equation

formula

Like a global expression, a **formula** is a (grammatically correct) assemblage of symbols *with at least* one occurence of *at least one* global variable but, unlike a global expression, *with* a calculus verb such as $=$, $<$, $\geqq$, $\ldots$.
(https://en.wikipedia.org/wiki/Well-formed_formula)

You may want to look up an old, classic game, WFF 'N PROOF. See (https://en.wikipedia.org/wiki/WFF_'N_PROOF.) which, by now is at the National Museum of American History but still available here and there on the web.

EXAMPLE 0.22.

- $-2.73 \ominus +13.22$ is *not* a formula,
- $-48.91 \geqq +33.1 \otimes x$ is a formula,
- $-1.0 \oplus +1.0 = 0$ is *not* a formula,
- $\dfrac{x^{+2} \ominus +7}{x \oplus +3} < 0$ is a formula

Formulas and global expressions are of course closely related.

EXAMPLE 0.23. If the variable x refers to the diameter of a circle, then the global expression in terms of x

$$3.14159 \times x$$

refers to the circumference of the circle.
On the other hand, if the variable y refers to the circumference of a circle, then the formula

$$y = 3.14159 \times x$$

relates the circumference with the diameter.

2. Sentences. Formulas are also what calculus sentences are constructed from. What complicates matters, though, is that, as will now be seen, there are completely different kinds of constructions which result in completely different kinds of sentences.

LANGUAGE 0.6 Unfortunately, the more or less standard words for these different kinds of sentences are not very evocative. (https://en.wikipedia.org/wiki/First-order_logic#Metalogical_properties.)

So, *RBC* will now introduce perfectly non-standard, but hopefully much more evocative, calculus words to denote these differently constructed kinds

numerical sentence
proposition
universal sentence
exisential sentence

of sentences together with LANGUAGE NOTES for the reader who wants to read more elsewhere.

https://en.wikipedia.org/wiki/Sentence_(mathematical_logic)

A. Numerical sentences. Very much as, when the variable in a global expression is replaced by a given number the result is an individual expression with a numerical value, when the variable(s) in a formula are *all* replaced by given number(s), the result is a **numerical sentence** with a truth value.

EXAMPLE 0.24. In EXAMPLE A.2 (Page 227),

- $-48.91 \geqq +33.1 \otimes x$ is *not* a numerical sentence,
- $-1.0 \oplus +1.0 \neq 0$ is a (FALSE) numerical sentence,
- $\dfrac{x^{+2} \ominus +7}{x \oplus +3} < 0$ is *not* a numerical sentence.

LANGUAGE 0.7 *RBC* calls such sentences numerical sentences because numerical sentences are constructed from numbers but numerical sentences usually go under the word **proposition** which, however, is more general in that, for instance, "Socrates is a man" is a proposition—but obviously *not* a numerical sentence.

B. Universal sentences. A **universal sentence** constructed from a formula says that *all* the numerical sentences obtained by replacing the variable by a given number are TRUE.
We employ $\forall \ulcorner \quad \urcorner$ with the formula written between $\ulcorner$ and $\urcorner$ and with the variable in the formula copied right after the symbol $\forall$.

EXAMPLE 0.25. To say that "$+1$ times *any* given number equals that number" , we employ the formula

$$+1 \odot x = x$$

together with the symbol $\forall \ulcorner \quad \urcorner$ to write the universal sentence

$$\forall x \ulcorner +1 \odot x = x \urcorner$$

which we read as:

For all x, $+1$ times x equals x

C. Existential sentences. An **exisential sentence** constructed from a formula says that *at least one* of the numerical sentences obtained by replacing the variable by a given number is TRUE.

We employ ∃⌜ ⌝ with the formula written between ⌜ and ⌝ and with the variable in the formula copied right after the symbol ∃.

Zero
non-zero number
0
0

EXAMPLE 0.26. To say that "−3.25 times *some* number equals˘ + 54.77", we employ the formula

$$-3.25 \odot x = +54.77$$

together with the symbol ∃⌜ ⌝ to write the existential sentence

$$\exists x \ulcorner -3.25 \odot x = x \urcorner$$

which we read as:

There exists at least one x such that -3.25 times x equals $+54.77$

6 Zero And Infinity

CAUTION 0.5 Somewhat unfortunately, the word 'zero' is employed in MATHEMATICS for two very different things:

- Something number-like in which case *RBC* will normally use symbols and, if the word is needed, will spell it 'Zero', with an upper-case 'Z'. Zero will be introduced and discussed in Subsection 6.2 - Infinity (Page 24) just below.
- As a feature functions may or may not have in which case *RBC* will spell it 'zero', with a lower-case 'z'.

1. Zero. Although totally and absolutely indispensable, already "*the ancient Greeks* [...] *seemed unsure about the status of zero as a number*[22].

CAUTION 0.6 Mathematicians distinguish ℕ, the whole numbers *including* 0, and ℕ*, the whole numbers *excluding* 0, AKA counting numbers, [a]

[a] https://en.wikipedia.org/wiki/Natural_number

When we first learned how to count, we always started with one, never with zero.

A. Semantics. The semantic question about **Zero** is simply: as opposed to **non-zero numbers**, that is *all* numbers except Zero, exactly what does Zero denote in the real world?

- Whether or not **0** is considered to be a whole number, there is no difficulty with **0** being whole number-*like* because, in the discrete aspect of

[22] https://en.wikipedia.org/wiki/0#Classical_antiquity

empty collection
0.
nothingness

the real world, there is no difficulty with collections with 0 item, that is with **empty collections**[23].

EXAMPLE 0.27. After we have eaten the last **apple** in a basket, the basket is empty and there is 0 **apple** in the basket.

- The difficulty is with **0.**, that is with 0 as decimal number-*like*, because, on the continuous side of the real world, there is no such thing as **nothingness** and thus no such thing as a 0. amount of stuff.

EXAMPLE 0.28.

▶ After we have drunk the last drop *in* a glass of **milk**, the glass is empty but needs to be washed because there still remains **milk** *on* the glass.

just as,

▶ There is no such thing as a perfect vacuum[a].
▶ There is no such thing as an absolute zero temperature[b]

[a] https://en.wikipedia.org/wiki/Vacuum
[b] https://en.wikipedia.org/wiki/Absolute_zero

But, even though Zero does not denote any entity, if only for convenience we will have to accept that

CAUTION 0.7 Zero *is* a *number*, albeit a *dangerous* number, that can therefore be a *given* number.

Of course, saying that the size of 0 is 0 merely moves the issue to plain *numbers.*

Isn't playing it both ways convenient?

B. Syntactics What complicates matters with Zero is that, from the syntactic viewpoint, the role Zero plays is complicated:

▶ Zero is less than *any* positive number and more than any negative number.

▶ With addition and subraction, Zero has much to do with opposite numbers:

- Adding Zero to a number results in the *same* number and adding two *opposite* numbers results in Zero,
- Subtracting Zero from a number results in that *same* number x_1 but subtracting a number from Zero results in the *opposite* number and subtracting a number from itself results in Zero.

▶ With multiplication, things are less satisfactory because, while:

[23] https://en.wikipedia.org/wiki/Natural_number

Which we tend to take for granted.

- multiplying two numbers by a *positive* number *keeps* the way the two numbers compare,

and that

- multiplying two numbers by a *negative* number *flips* the way the two numbers compare,

the danger is to forget that

- multiplying two numbers by zero *destroys* the way the two numbers compare because the result is Zero = Zero.

▶ But it's with division that things get *really* bad:

- Dividing Zero by any *non-zero* number results in Zero no matter what.

EXAMPLE 0.29. $0 \div 3 = 0$ because, when we share in the real-world 0 **apples** among 3 **persons** nobodybody gets any **apple**:

$$\frac{0\,\textbf{apple}}{3\,\textbf{persons}} = \frac{3 \times 0\,\textbf{apple}}{3 \times 1\,\textbf{person}} = \frac{\cancel{3} \times 0\,\textbf{apple}}{\cancel{3} \times 1\,\textbf{person}} = \frac{0\,\textbf{apple}}{1\,\textbf{person}} = 0\,\textbf{apples}/\textbf{person}$$

which we can check as follows

$$0\,\textbf{apples}/\textbf{person} \times 3\,\textbf{persons} = \frac{0\,\textbf{apple}}{1\,\cancel{\textbf{person}}} \times 3\,\cancel{\textbf{persons}} = 0\,\textbf{apple} \times \frac{3}{1} = 0\,\textbf{apples}$$

And, worst of all,

- Dividing a non-zero number by Zero just cannot be done.

EXAMPLE 0.30. When we divide 12 **apples** among 3 **persons** each **person** gets 4 **apples** and altogether we hand out 12 **apples**:

$$4\,\textbf{apples}/\textbf{person} \times 3\,\textbf{persons} = \frac{4\,\textbf{apples}}{1\,\cancel{\textbf{person}}} \times 3\,\cancel{\textbf{persons}} = 4\,\textbf{apples} \times \frac{3}{1} = 12\,\textbf{apples}$$

but, we cannot divide 12 **apples** among 0 **person** because, whatever each **person** gets, $?$ **apples**/**person**, we can only hand out 0 **apples**:

$$?\,\textbf{apples}/\textbf{person} \times 0\,\textbf{persons} = \frac{?\,\textbf{apples}}{1\,\cancel{\textbf{person}}} \times 0\,\cancel{\textbf{persons}} = ?\,\textbf{apples} \times \frac{0}{1} = 0\,\textbf{apple}$$

which, among other things, can prevent evaluating a global expression AT a given number.

EXAMPLE 0.31. See step **c.** in DEMO 0.2b (Page 16) and DEMO 0.2c (Page 17)

Thus,

infinity
endless
end of the line

2. Infinity. Contrary to Zero, **infinity** is not necessary for ARITHMETIC but, as we will see, just as totally and absolutely indispensable for CALCULUS.

But, already way back, and a lot more than Zero, infinity has been a nightmare: "*Since the time of the ancient Greeks, the philosophical nature of infinity was the subject of many discussions among philosophers.*" [24]

A. Semantics The question about infinity is the same as with zero: what does infinity denote in the real world?

a. But with infinity, there is already a difficulty in the discrete aspect of the real world in that there is no such entity in the real world as a collection with an infinity of items.

EXAMPLE 0.32. There is no infinity of stars in the universe, only a hugely huge number of stars. Beyond our ability even to imagine, certainly, infinite, no.
And, yes, there is an infinity of whole numbers but whole numbers are not real world entities.

b. And, in the continuous aspect of the real world, things are much worse. As Leibniz said, "*There are two labyrinths of the human mind: one concerns the composition of the continuum, and the other the nature of freedom, and both spring from the same source—the infinite.*"

To begin with, there is no such thing in the real world as an infinite amount of stuff.

EXAMPLE 0.33. The amount of energy in the universe is not infinite, only hugely huge. Beyond our ability even to imagine, yes, infinite, no.

And then, while it seems that, say, length of travel, could be **endless**, when we actually do try to go farther and farther away, even though we have the feeling that the longer we go, the farther away we will get, and that there is nothing to keep us from getting as far away as we want, ln the real world there is no such thing as endlessness in that, sooner or later, we get to the **end of the line**

B. Syntactics Here it is better not even to attempt calculating with infinity. But the curious reader might want to see Subsection 7.3 - Extended numbers (Page 26).

3. Are ∞ and 0 reciprocal? Another reason for *not* computing with infinity is that,

[24] https://en.wikipedia.org/wiki/Infinity

- From the *division* table, we get that $\frac{x_{\text{pos}}}{-\infty} = 0^-$ and therefore, in particular, that $\frac{+1}{-\infty} = 0^-$ so that, as would be expected, the reciprocal of $-\infty$ is 0^- and, similarly, we get that the reciprocal of $+\infty$ is 0^+,
- However, from the *multiplication* table we get only that $-\infty \odot 0^- =+$? and that $+\infty \odot 0^+ =+$?

upper end of the line
lower end of the line

While not contradictory, this would be annoying and, as we will see in THEOREM 0.4 - Otiming qualitative sizes (Page 41), we will have a much more satisfying way to compute whether or not 0 and ∞ are reciprocal.

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

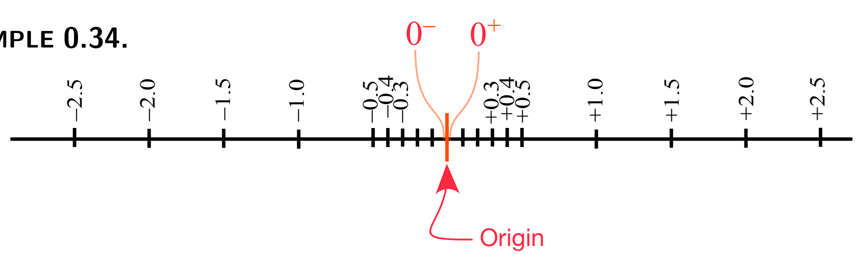
()

7 Compactifying Numbers

1. Numbers and zero. Zero corresponds to the origin of a ruler. But, inasmuch as there can be only so many digits in a signed decimal number, RBC will often employ

- **0^-** as **upper end of the line** for *negative* decimal numbers
- **0^+** as **lower end of the line** for *positive* decimal numbers

EXAMPLE 0.34.



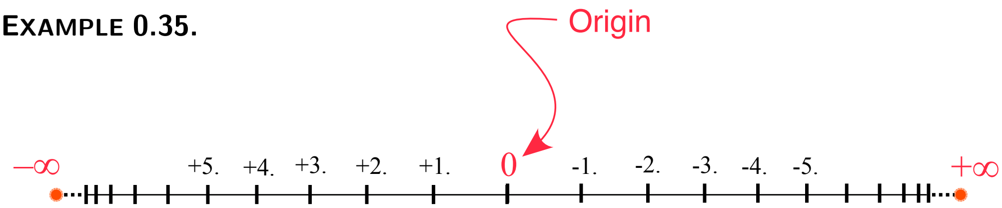
extended number

2. Numbers and infinity. Infinity corresponds to the ends of a ruler. But, inasmuch as there can be only so many digits in a signed decimal number, we will often use

- $-\infty$ as lower end of the line for *negative* decimal numbers.
- $+\infty$ as upper end of the line for *positive* decimal numbers

Here again, though, marks picturing "equidistant" numbers cannot themselves be "equidistant" and have to get closer and closer as the numbers get larger.

EXAMPLE 0.35.



3. Extended numbers. In fact, and while *RBC* will *not* do so, it is even possible to *compute*, at least to an extent, with the **extended numbers**, that is with 0^-, 0^- and $+\infty$, $-\infty$, together with the decimal numbers.

(https://en.wikipedia.org/wiki/Extended_real_number_line#Arithmetic_operations)

$\oplus$	$-\infty$	y_{neg}	0^-	0^+	y_{pos}	$+\infty$
$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	?
x_{neg}	$-\infty$	z_{neg}	x_{neg}	x_{neg}	?	$+\infty$
0^-	$-\infty$	y_{neg}	0^-	$0^?$	y_{pos}	$+\infty$
0^+	$-\infty$	y_{neg}	$0^?$	0^+	y_{pos}	$+\infty$
x_{pos}	$-\infty$	?	x_{pos}	x_{pos}	z_{pos}	$+\infty$
$+\infty$	?	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$

$\ominus$	$-\infty$	y_{neg}	0^-	0^+	y_{pos}	$+\infty$
$-\infty$	?	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
x_{neg}	$+\infty$	$z_?$	x_{neg}	x_{neg}	z_{neg}	$-\infty$
0^-	$+\infty$	y_{pos}	$0^?$	0^-	y_{neg}	$-\infty$
0^+	$+\infty$	y_{pos}	0^+	$0^?$	y_{neg}	$-\infty$
x_{pos}	$+\infty$	z_{pos}	x_{pos}	x_{pos}	$z_?$	$-\infty$
$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	?

two-point compactification

$\odot$	$-\infty$	y_{neg}	0^-	0^+	y_{pos}	$+\infty$
$-\infty$	$+\infty$	$+\infty$	$+?$	$-?$	$-\infty$	$-\infty$
x_{neg}	$+\infty$	z_{pos}	0^+	0^-	z_{neg}	$-\infty$
0^-	$+?$	0^+	0^+	0^-	0^-	$-?$
0^+	$-?$	0^-	0^-	0^+	0^+	$+?$
x_{pos}	$-\infty$	z_{neg}	0^-	0^+	z_{pos}	$+\infty$
$+\infty$	$-\infty$	$-\infty$	$-?$	$+?$	$+\infty$	$+\infty$

$\oslash$	$-\infty$	y_{neg}	0^-	0^+	y_{pos}	$+\infty$
$-\infty$	$+?$	$+\infty$	$+\infty$	$-\infty$	$-\infty$	$-?$
x_{neg}	0_{pos}	z_{pos}	$+\infty$	$-\infty$	z_{neg}	0_{neg}
0^-	0^+	0^+	$+?$	$-?$	0^-	0^-
0^+	0^-	0^-	$-?$	$+?$	0^+	0^+
x_{pos}	0^-	z_{neg}	$-\infty$	$+\infty$	z_{pos}	0^+
$+\infty$	$-?$	$-\infty$	$-\infty$	$+\infty$	$+\infty$	$+?$

One reason we will *not* compute with extended numbers is of course the yellow boxes in the above operation tables.

4. Compactifications. [25] What shape is the real world is a very serious question in ASTROPHYSICS. (https://www.quantamagazine.org/what-is-the-geometry-of-the-universe-20200316/.)

1. If the real world is flat, this means that the numbers are pictured by rulers. But then, there may or may not be an end of the line to rulers. If there is, then the extended numbers that is the numbers together with $+\infty$, $-\infty$ as well as 0^-, 0^- make up what is called a **two-point compactification** of the signed decimal numbers.

2. On the other hand, if the real world is not flat, there may be *no* end of the line and, after a long journey, we may find ourselves back to where we started. (https://www.quantamagazine.org/what-shape-is-the-universe-closed-or-flat-20191104

EXAMPLE 0.36. Even though Magellan died in 1521 while trying to go as far away from Seville as he could[26], his ships kept on going west.

[25] https://www.cantorsparadise.com/two-compactification-theorems-6a73b11ea908

Magellan circle
one-point compactification
origin

Bearing witness that there was no going around the fact that the earth is round.

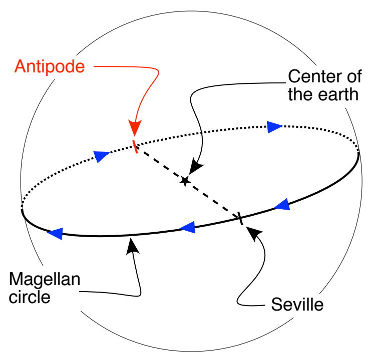
And one of his ships eventually reached . . . home.

https://en.wikipedia.org/wiki/Ferdinand_Magellan#Voyage

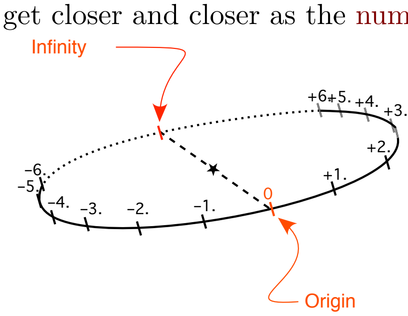
(https://www.cantorsparadise.com/two-compactification-theorems-6a73b11ea908)

In that case, since what looks to us like a *straight line* is in the real world just a piece of a **Magellan circle**, instead of rulers we will often employ **one-point compactifications** of the numbers, that is Magellan circles that include ∞, "down under" from the origin, together with the numbers.

origin, as end of the line. together with the numbers

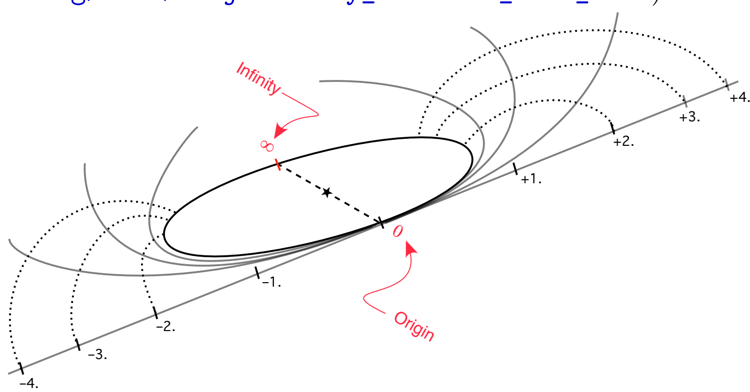


Of course, marks picturing "equidistant" numbers cannot themselves be "equidistant" and have to get closer and closer as the numbers get larger:



Another way to look at this is to imagine bending the extremities of the ruler, while shrinking the ends more and more, until they meet. (https:

//en.wikipedia.org/wiki/Projectively_extended_real_line)



8 Size Of Numbers

While the ordinary English verbs, *is-larger-than*, *is-smaller-than*, and *is-the-same-as*, all take the sign as well as the size into account just like the corresponding calculus verbs, *is-more-than*, *is-less-than is-equal-to*, the ordinary English adjectives, *large*, *small* and *medium* refer only to the size and not to the sign.

EXAMPLE 0.37. Essentially all dictionaries define *large* as "bigger than usual in size":
- "*exceeding most other things of like kind especially in quantity or size*"[a]
- "*of greater than average size*"[b] "*of more than average size*"[c]
- "*of more than average size*"[d]
- "*greater in size than usual or average*" [e]
- "*Of considerable size or extent; great, big. Designating a quantity, amount, measure, etc., of relatively great magnitude or extent.*"[f]

[a] https://www.merriam-webster.com/dictionary/large
[b] https://www.thefreedictionary.com/Large
[c] https://www.dictionary.com/browse/large
[d] https://www.dictionary.com/browse/large
[e] https://www.collinsdictionary.com/dictionary/english/large
[f] https://www.oed.com/search/dictionary/?scope=Entries&q=large

1. Size-comparing signed numbers. In order to define calculus adjectives that correspond to the ordinary English adjectives, *large*, *small* and

size-comparie
smaller-size
larger-size
equal-in-size

medium, it will be convenient to define **size-comparing** which is comparing in terms of only the *sizes* of the signed numbers and ignoring the *signs* :

DEFINITION 0.4 Given two signed numbers x and y,
- x is **smaller-size** than y iff Size $x <$ Size y,
- x is **larger-size** than y iff Size $x >$ Size y,
- x is **equal-size** to y iff Size $x =$ Size y, (So, iff x and y are either equal or opposite.)

EXAMPLE 0.38. To size-compare -254.7 and-32.6:
Since:

The *size* of -254.7 is 254.7 and the *size* of $+32.6$ is 32.6,

and since

$$254.7 > 32.6$$

then

The *size* of -254.7 $>$ the *size* of $+32.6$

In other words, i

-254.7 is larger-size than $+32.6$,

even though -234.7 is smaller than $+32.6$

EXAMPLE 0.39. To size-compare $+71.44$ and -128.52:
Since

the size of $+0.7$ is 0.7 and the size of -128.52 is 128.52

and since

71.44 is smaller than128.52

then

The *size* of $+71.44$ is smaller than the *size* of -128.52

that is

$+71.44$ is smaller-size than -128.52,

even though $+71.44$ is larger than -128.52

EXAMPLE 0.40. To size-compare -0.7 and$+0.7$:
Since

the size of -0.7 is 0.7 and the size of $+0.7$ is 0.7,

and since

0.7 is equal to 0.7

then

The *size* of -0.7 is equal to the *size* of $+0.7$

that is

-0.7 is equal-size to $+0.7$,

even though -0.7 is smaller than $+0.7$

closer
farther

CAUTION 0.8 There are *no symbols* for size-comparisons of numbers.

In fact, because size-comparing is *not* standard, size-comparing is invariably confused with comparing sizes but:

CAUTION 0.9 Given two signed number-phrases,
- *Comparing* the number-phrases results in a statement about the numerators

while
- *Size*-comparing the number-phrases results in a statement about the denominators.

EXAMPLE 0.41. Let **Dick** be 13 and **Jane** be 18.
Then,
- *Comparing the ages* of **Dick** and **Jane** is talking about the *ages* of **Dick** and **Jane** that is talking about *numbers* namely: "$13 < 18$",
- *Age-comparing* **Dick** and **Jane** is talking about **Dick** and **Jane** *themselves* that is talking about *people* namely: "**Dick** *is younger than* **Jane**".

EXAMPLE 0.42. Let Dick be worth $+13,000$ **Dollars** and Jane be worth $-128,000,000$ **Dollars**.
Then,
- *Comparing* the worths of Dick and Jane, that is comparing $+13,000$ and $-128,000,000$ which shows that Dick is richer than Jane,

but
- *Size-comparing* the worths of Dick and Jane, that is comparing $13,000$ and $128,000,000$ which shows that Dick is a lot less important a person than Jane is.

When picturing size-comparisons of two given numbers
- The smaller-size number is **closer** to 0 than the larger-size number,
- The larger-size number is **farther** from 0 than the smaller-size number.

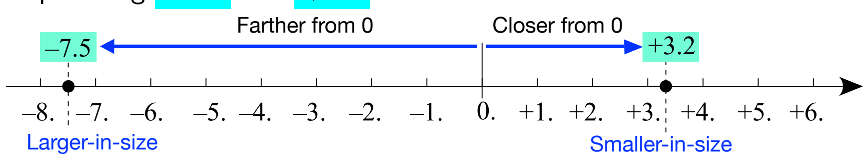
EXAMPLE 0.43. Given the numbers -7.5 and $+3.2$, we saw in EXAMPLE 1.9 (Page 73) that

- -7.5 is *larger-size-than* $+3.2$,

and therefore that

- $+3.2$ is *smaller-size-than* -7.5,

After picturing -7.5 and $+3.2$



we see that

- -7.5 is *farther from* 0 than $+3.2$,
- $+3.2$ is *closer from* 0 than -7.5,

In particular:

THEOREM 0.1 Sizes of *reciprocal* numbers:
- The larger-size a non-zero number is, the smaller-size its reciprocal, and
- The smaller-size a non-zero number is, the larger-size its reciprocal.

Getting there, eh?

Proof. zzzzz □

But, even though we all have an *intuitive* idea of what the ordinary English words *large*, *small* and *medium* mean, the numbers to which the adjectives *large*, *small* and *medium* apply are not necessarily the same in all situations.

EXAMPLE 0.44. Nobody likes to work for a *small* amount of money but while billionaires would say a thousand dollars an hour is way too small to even dream of, the rest of us would probably think a hundred dollars an hour large enough.

Of course, in some parts of the real world, even a dollar an hour is actually a large amount of money.

2. Giveable numbers. We can of course give *any* signed decimal number we want but there are unbelievably many numbers that are unbelievably larger-size than any number you care to imagine as well as unbelievably many numbers that are unbelievably smaller-size than any number you care to imagine:

- We all went through a stage as children when we would count, say, "*one, two, three, twelve, seven, fourteen, ...*" but soon after that we were able to count properly and then we discovered that there was no largest number: we could *always* count one more. (Of course, counting backwards into the negative numbers has no end either so there is no largest-size number.) But that was only the tip of the iceberg.

EXAMPLE 0.45. Start with, say -73.8, and keep multiplying by 10 by moving the decimal point to the *right*, inserting **0**s *left* of the decimal point when it becomes necessary

-73.8
-738.
-7 38**0**.
-73 80**0**.
-738 00**0**.
-7 380 00**0**.
...
-738 000 000 000 000 000 000 000 000 000 00**0**.

This last number is probably already a lot larger-size than any number you are likely to have ever encountered.

*If not, just keep inserting **0**s until you get there!*

(See https://en.wikipedia.org/wiki/Large_numbers#Large_numbers_in_the_everyday_world)

- On the other hand, as children knowing only *plain* whole numbers, we thought there was no number smaller than 1 or perhaps than 0. With decimal numbers, though, there is no smallest-size number.

EXAMPLE 0.46. Start with, say 41.6, and keep dividing by 10 by moving the decimal point to the *left*, inserting **0**s *right* of the decimal point when it becomes necessary.

size-range

41.6
4.16
0.416
0.041 6
0.004 16
0.000 416
0.000 041 6
0.000 004 16
...
0.000 000 000 000 000 000 000 000 000 000 000 000 004 16

If not, keep inserting 0s until you get there!

This last number is probably already a lot smaller-size than any number you are likely to have ever encountered in a real world situation.

What numbers *scientists* and *engineers* use, though, fall into **size-ranges** that depend on the situation.

EXAMPLE 0.47. The numbers that *astrophysicists*[a] give and the numbers that *nanophysicists*[b] give definitely fall into entirely different size-ranges.

[a] https://en.wikipedia.org/wiki/Astrophysics
[b] https://en.wikipedia.org/wiki/nanophysicist

In this regard, the following, all about distances—which are sizes, are well worth looking up:

- The 9 minutes 1977 classic video by Charles Eames[a]
- The presentation by **Terence Tao** (Fields Medal 2006) [b].

[a] https://www.youtube.com/watch?v=0fKBhvDjuy0
[b] http://terrytao.files.wordpress.com/2010/10/cosmic-distance-ladder.pdf

Which is one reason why scientists and engineers employ ***metric*** *units of stuff: the conversion of metric units of stuff is easy because it involves only moving the decimal point without changing the digits. Scientists work hard enough not to bother with inconvenient units of stuff but many engineers in this country often have to use the US Customary System.*

Of course, units of stuff of the appropriate size allow us to use numbers in whatever size-range is convenient—

EXAMPLE 0.48. In the US Customary System,

- Instead of talking about 38016 inches, we usually say 0.6 miles,
- Instead of talking about 0.01725 tons, we usually say 34.5 pounds.

while, in the Metric System,

- Instead of talking about \$3 370 000., we usually say 3.37 MegaDollars.
- Instead of talking about 0.0000074 Meters, we usually say 7.4 microMeters.

Since +1.0 unit and −1.0 unit are most likely to be in any range,

out-of-range
cutoff-size
upper cutoff-size
lower cutoff-size
negative range
positive range
negative upper cutoff-number
negative lower cutoff-number
positive upper cutoff-number
positive lower cutoff-number

> **AGREEMENT 0.4** As far as *RBC* will be concerned, +1.0 and −1.0 will *always* be within the size-range.

By the same token, when we *use* numbers, for *scientists* and *engineers*,, in any real world situation there will be numbers **out-of-range**, that is numbers that we will *not* use.

EXAMPLE 0.49. Numbers like

a. −7 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000.

or

b. −0 .000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 000 003

are *not* very likely to be within any range.

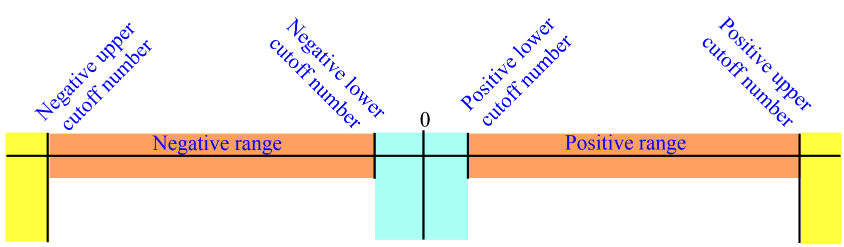
So, in the real world there always are two **cutoff-sizes** that determine the size-range:

- An **upper cutoff-size** *above* which numbers will surely *not* designate anything in the situation,
- A **lower cutoff-size** *below* which numbers will surely *not* designate anything in the situation.

Unfortunately, often left to go without saying.

In practice, though, it is more convenient to distinguish the **negative range** from the **positive range** and, instead of cutoff-sizes, employ:

- **Negative upper cutoff-number**
- **Negative lower cutoff-number**
- **Positive upper cutoff-number**
- **Positive lower cutoff-number**

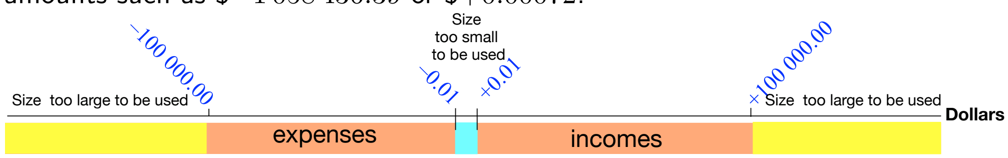


Of course, the cutoff-sizes will depend on the real world situations.

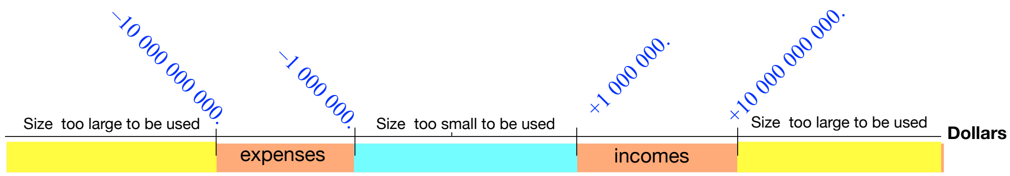
finite number

EXAMPLE 0.50. Some rulers show 1/32 inch, some tape measure show

EXAMPLE 0.51. A small business could take 100 000.00 and 0.01 as cutoff *sizes* for their accounting system as it probably would never have to deal with amounts such as $−1 058 436.39 or $+0.00072.



In contrast, the accounting system for a multinational corporation would certainly employ different cutoff-sizes, maybe something like:



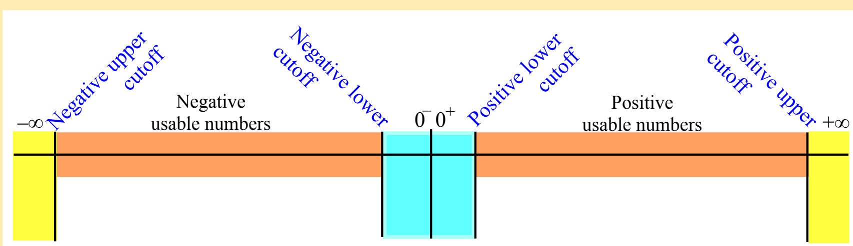
So, given a size-range, the numbers that the user can give are:

DEFINITION 0.5 **Finite numbers**[a] are numbers that are in the size-range that is *both*

smaller-size-than the *upper* cutoff-size

and

larger-size-han the *lower* cutoff size



[a] https://en.wikipedia.org/wiki/Finite_number)

Both +1 and −1 are finite numbers since +1 and −1 correspond to *units* of stuff.

infinitesimal number
small variable
h

THEOREM 0.2 Finite numbers are *non-zero* numbers.(But non-zero numbers are *not necessarily* finite numbers.)

Proof. Acording to **?? ??** - **??** (**??**) and as the represent illustrates,

- The upper cutoff-size keeps finite numbers away from $-\infty$ and $+\infty$.
- The lower cutoff-size keeps finite numbers away from 0^- and 0^+.

□

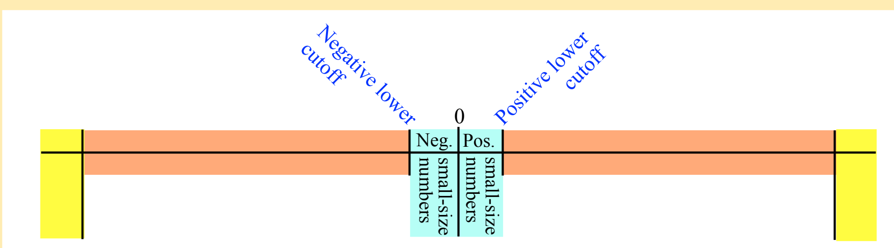
AGREEMENT B.1 (Restated) 'Number' (without qualifier) Finite number will be short for *reasonable* signed decimal numbers in the given size-range

RBC will employ the generic given number symbols x_0 , x_1 , $x_2, \ldots$ as variable for finite numbers.

3. Off-range numbers. While off-range numbers cannot be finite numbers, off-range numbers actually play a big role in Calculus and *RCB* will employ the following names for off-range numbers:

A. Infinitesimal numbers. The numbers whose size is too *small* for the numbers to be *giveable* will be referred to in *RBC* as:

DEFINITION 0.6 Given a size-range, **infinitesimal numbers**[a] are numbers that are smaller-size than the *lower* cutoff-size



[a]https://en.wikipedia.org/wiki/Infinitesimal

DEFINITION 0.7 The **small variables** $\boldsymbol{h}$, $\boldsymbol{k}$, ... will be the (standard) symbols for infinitesimal numbers.

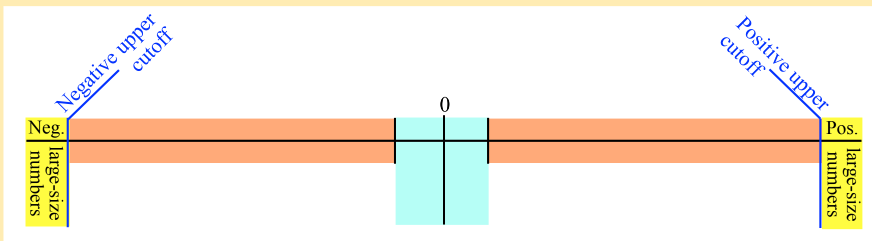
near-zero number
infinite number
large variable
L
M
infinite number
near-infinity number

CAUTION 0.10 because 0 has *no* size to begin with. (**??** **??** - **??** (**??**))

Also known as **near-zero numbers**.

B. Infinite numbers. The numbers whose size is too *large* for the numbers to be *giveable* will be referred to in *RBC* as:

DEFINITION 0.8 Infinite numbers[a] are numbers that are larger-size than the *upper* cutoff-size



[a] https://www.dictionary.com/browse/infinite)

DEFINITION 0.9 The **large variables** $\boldsymbol{L}$, $\boldsymbol{M}$, ... will be the (standard) symbols for infinite numbers.

CAUTION 0.11 ∞ is *not* a number to begin with. (CAUTION 0.2 - No other number (Page 5))

Infinite as in out of bound. But then so is zero.

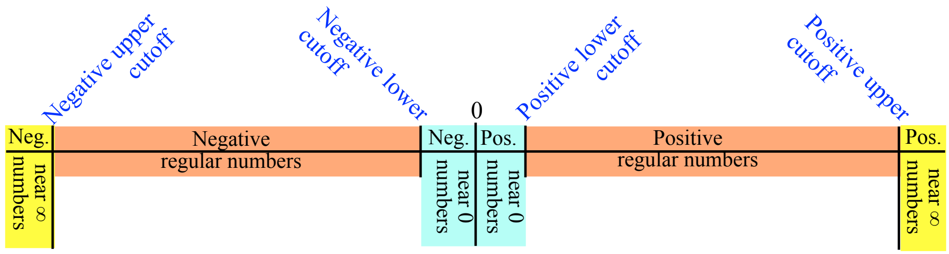
Also known as **near-infinity numbers** .

========================

========================

While the variables x, y, z can stand for numbers of *any* qualitative sizes,

Altogether, then, these qualitative sizes are illustrated by:



REWRITE ALL THIS SECTION USING h and L

In ARITHMETIC, we calculate in exactly the same way with *all* signed decimal numbers), regardless of their size.

EXAMPLE 0.52. $+0.3642$ and -105.71 are added, subtracted, multiplied and divided by exactly the same rules as $-41\,008\,333\,836\,092.017$ and -0.000001607.

=======Begin WORK ZONE=======

While 0 does not exist in the real world, infinitesimal numbers do exist in the real world

h^n

So, while $5 \oplus 0$ does not exist in the real world so that we do not want to write $5 \oplus 0 = \infty$, infinitesimal number does exist in the real world and there is no problem writing $5 \oplus h = L$ /Users/alainschremmer/Desktop/untitled folder infinitesimal number $\oplus$ infinitesimal number

========End WORK ZONE=======

For *calculating* purposes, qualitative sizes make up a rather crude system because qualitative sizes carry no information whatsoever about *where* the cutoffs are.

Nevertheless, as we will see, the calculations we *can* do with qualitative sizes will be plenty enough to help us simplify calculations by separating what is qualitatively the right size to be relevant to what we are interested in from what is qualitatively the wrong size and therefore irrelevant to what we are interested in.

And if you're worried about rigor, you'll be glad to know qualitative sizes *lead straight to Bachmann-Landau's little o's and big O's (https://en.wikipedia.org/wiki/Big_O_notation).*

We will now discuss to what extent we can calculate with *numbers* of which all we know is their qualitative size: infinite, or infinitesimal, or medium-size.

In each case, it is most important that you develop a good feeling for what is happening and so it is important for you to experiment by setting cutoff-sizes and then picking numbers with the qualitative sizes you want. A good rule of thumb for picking:

You don't need extreme *cutoff-sizes but do pick your* numbers far *from the cutoffs.*

undetermined

- medium-size numbers is to try ± 1,
- infinite numbers is to try ± 10.0 or ± 100.0 or ± 1000.0 etc
- infinitesimal numbers is to try ± 0.1 or ± 0.01 or ± 0.001 etc

4. Adding and subtracting qualitative sizes.

THEOREM 0.3 Oplussing qualitative sizes numbers

$\oplus$	near ∞	regular	near 0
near ∞	**?**	near ∞	near ∞
regular	near ∞	**?**	regular
near 0	near ∞	regular	near 0

In other words

$\oplus$	L	regular	h
near ∞	**?**	L	L
regular	L	**?**	regular
h	L	regular	h

Proof. **i.** The non-highlighted entries are as might be expected.

EXAMPLE 0.53. $-100\,000 \oplus +1\,000 = -99\,000$
$-100\,000 \oplus -0.001 = 100\,000.001$

So, the reader is invited to decide on cutoff-sizes, experiment a bit, and then prove the non-highlighted entries using these cutoff-sizes.

ii. When the two infinite numbers have opposite signs, the addition is **undetermined** because the result could then be infinite, or infinitesimal, or medium-size, depending on "how much" infinite the two infinite numbers are compared to each other.

EXAMPLE 0.54. Here are two additions of infinite numbers whose results are different in *qualitative sizes*:
$+1\,000\,000\,000\,000.7 \oplus -1\,000\,000\,000.4 = +999\,000\,000\,000.3$,
but
$-1\,000\,000\,000\,000.5 \oplus +1\,000\,000\,000\,000.2 = -0.3$.

□

=======Begin WORK ZONE=======

Since $\ominus = \oplus$ Opposite

=======End WORK ZONE========

5. Multiplying qualitative sizes.

THEOREM 0.4 Otiming qualitative sizes

$\odot$	infinite	medium-size	infinitesimal
infinite	infinite	infinite	?
medium-size	infinite	medium-size	infinitesimal
infinitesimal	?	infinitesimal	infinitesimal

The global symbols have different subscripts because, even when they have the same qualitative size, they stand for different numbers.

Proof. **i.** The non-highlighted entries are as might be expected.

EXAMPLE 0.55.

$$-10\,000 \odot -1\,000 = +10\,000\,000$$
$$+0.01 \odot -0.001 = -0.00001$$

So, the reader is invited to decide on cutoff-sizes, experiment a bit, and then prove the non-highlighted entries using these cutoff-sizes.

ii. infinite $\odot$ infinitesimal is undetermined because the result could be infinite, or infinitesimal, or medium-size, depending on "how much infinite" infinite is compared to "how much infinitesimal" infinitesimal is.

EXAMPLE 0.56. Here are different instances of infinite $\odot$ infinitesimal that result in different *qualitative sizes*:

$-1\,000 \odot -0.1 = +100$	$-100\,000\,000 \odot -0.00\,001 = +100$
$+1\,000 \odot -0.001 = -1$	$+1\,000\,000 \odot -0.00\,001 = -1$
$+1\,000 \odot +0.00\,001 = +0.01$	$+1\,000 \odot +0.00\,001 = +0.01$

Similarly for infinitesimal $\odot$ infinite.

□

6. Dividing qualitative sizes.

THEOREM 0.5 Odividing qualitative sizes

$\frac{\quad}{\quad}$	infinite	medium-size	infinitesimal
infinite	?	infinite	infinite
medium-size	infinitesimal	medium-size	infinite
infinitesimal	infinitesimal	infinitesimal	?

The global symbols have different subscripts because, even when they have the same qualitative size, they stand for different numbers.

Proof. **i.** The non-highlighted entries are as might be expected.

EXAMPLE 0.57. $\frac{-10\,000\,000}{+50} = -200\,000$
$\frac{+0.03}{+6\,000\,000} = +0.000\,000\,005$

So, the reader is invited to decide on cutoff-sizes, experiment a bit, and then prove the non-highlighted entries using these cutoff-sizes.

ii. $\frac{\text{infinite}}{\text{infinite}}$ is undetermined because the result could be infinite, or infinitesimal, or medium-size, depending on "how much infinite" infinite and infinite are compared to each other..

EXAMPLE 0.58. Here are three instances of $\frac{\text{infinite}}{\text{infinite}}$ that result in different *qualitative sizes*:
$\frac{-1\,000\,000}{-1\,000} = +1\,000$, $\frac{-1\,000\,000}{-100\,000} = -10$, $\frac{-100\,000}{-1\,000\,000\,000} = +0.000\,1$.

And $\frac{\text{infinitesimal}}{\text{infinitesimal}}$ is similarly undetermined.

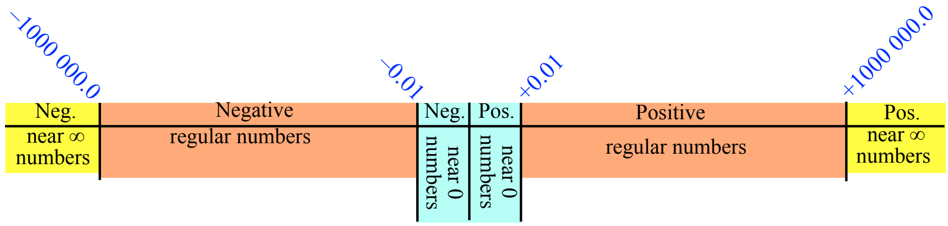
EXAMPLE 0.59. Here are three instances of infinitesimal $\oplus\!\!\!\div$ infinitesimal that result in different *qualitative size*:
$-0.001 \div +0.1 = -0.01$, $+0.001 \div +0.001 = +1$, $-0.01 \div -0.001 = +10$

□

7. Reciprocal of a qualitative size. We really would like the reciprocal of a infinitesimal number to be a infinite number and, the other way round, the reciprocal of a infinite number to be a infinitesimal number.

i. Unfortunately, because we defined qualitative sizes in terms of cutoff-sizes which we decide independently of each other, this is *not necessarily* the case and the reciprocal of a infinitesimal number need *not* be a infinite number and, the other way round, the reciprocal of a infinite number need *not* be a infinitesimal number because the upper cutoff-size and the lower cutoff-size are *not necessarily* reciprocal of each other.

EXAMPLE 0.60. The following cutoff-sizes are probably suitable for the accounting system of a small business:



i. $+0.009$ is below the positive lower cutoff ($+0.009 < +0.01 = +0.010$) and is therefore a infinitesimal number,
ii. The reciprocal of $+0.009$ is $+111.1$ (Use a calculator.)
iii. $+111.1$ is below the positive upper cutoff and is therefore *not* a infinite number.

ii. Fortunately, it is always possible to take the cutoff-sizes so that

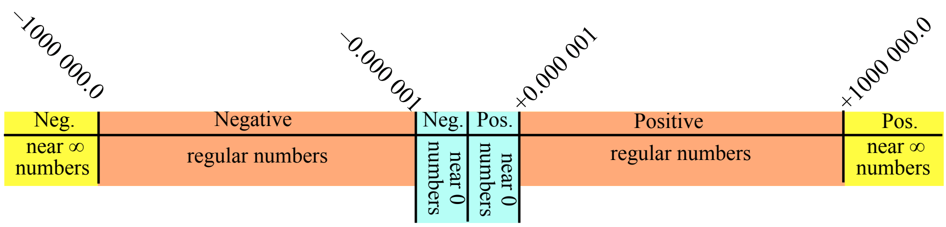
- the upper cutoff-size *is* the reciprocal of the lower cutoff-size

and, the other way round,

- the lower cutoff-size *is* the reciprocal of the upper cutoff-size

because all that will happen is that with the adjusted cutoff-sizes there will now be more numbers that will be medium-size than is really needed.

Example 0.61. We can change the lower cutoff-size in ?? (??) to $0.000\,001$:



so that now the lower cutoffs and the upper cutoffs are reciprocal of each other:
i. $+0.000\,9$ is below the positive lower cutoff ($+0.0009 < +0.001 = +0.0010$) and is therefore a infinitesimal number,
ii. The reciprocal of $+0.0009$ is $+1\,111.1$ (Use a calculator.)
iii. $+1\,111.1$ is above the positive upper cutoff and is therefore a infinite number.
The price is just thatnumbers whose size is between 0.01 and 0.000001 will now also be medium-size—but most probably will never be used.

iii. So then, from now on,

Agreement 0.5 The lower cutoff-size and the upper cutoff-size will be reciprocal of each other.

iv. We then have:

Theorem 0.6 Reciprocity of qualitative sizes

- $$\text{Reciprocal of infinite number} = \frac{+1}{\text{infinite number}} = \text{infinitesimal number}$$
- $$\text{Reciprocal of infinitesimal number} = \frac{+1}{\text{infinitesimal number}} = \text{infinite number}$$
- $$\text{Reciprocal of medium-size number} = \frac{+1}{\text{medium-size number}} = \text{medium-size number}$$

Proof.

- If a given number is infinite,
  - By Definition 0.4 - Size-comparison (Page 30), the given number is larger-size than the upper cutoff-size
  - By Theorem A.1 - Opposite numbers add to 0: (Page 227), the reciprocal of the given number is then smaller-size than the reciprocal of the upper cutoff-size.
  - But by Agreement A.1 - Computable expressions (Page 228), the reciprocal of the upper cutoff-size is the lower cutoff-size.
  - So, the reciprocal of the given number is smaller-size than the lower cutoff-size.
  - And so, by Definition 0.4 - Size-comparison (Page 30), the reciprocal of the given infinite number is a infinitesimal number
- The reader is invited to make the case for the reciprocal of a infinitesimal given.
- The reader is invited to make the case for the reciprocal of a medium-sizegiven number that is medum-size

□

9 Neighborhoods - Local Expressions

This is where Calculus parts away from Discrete Mathematics .

1. Points. In spite of **?? ??** - **??** (**??**) and Caution 0.2 - No other number (Page 5), and because, for all their differences, we will be *using* 0, ∞, and non-zero numbers pretty much in the same way, it will be extremely convenient to employ a word to stand for any of 0 or ∞ as well as for x_0:

point
nearby number
near 0

Definition 0.10 By **point**, we will mean any of the following:

- *Any* non-zero number,
- 0, (Even though 0 has no sign.)
- ∞. (Even though ∞ is *not* a number.)

Thus, a **given point** can be 0 as well as a non-zero number but can also be ∞.

In particular, it will be extremely convenient to see the *points* ∞ and 0 as points that are reciprocal of each other.

Nevertheless:

Caution 0.12 One cannot *compute* with points because the rules for computing with *non-zero* numbers and with 0 are different and we cannot compute with ∞ very much at all.

2. Nearby numbers. Evaluating a global expression *at* a point, though, is to ignore the real world and, in fact, since, as we will see in **?? ??** - **??** (**??**), Calculus deals with 'change', instead of wanting to investigate what happens *At* a given point, we will investigate what happens *At* **nearby numbers**.

In a crime novel, the victim is never the story. The story is always around *the victim. (Anonymous crime writer.)*

Example 0.62. As opposed to Example A.2 (Page 227), we can tell a car is moving from a *movie*, that is from still pictures during a short time span.

More precisely:

i. As we saw in Section 2 - Issues With Decimal Numbers (Page 6), *nothingness* does not exist in the real world,

Example 0.63. We employ 0 quart of milk to designate the amount of milk that appears to be in an empty bottle but it might just be that the amount of milk in the bottle is just too small for us to see.

Just how clean is clean?

So, in accordance with the real world, we will employ nearby numbers that is, in this case, numbers **near** 0, that is *infinitesimal* numbers,

near ∞
neighborhood
thicken
center

EXAMPLE 0.64. $-0.002.078$ and $+0.000.928$ are both near 0.

ii. As we saw in Section 3 - Giving Numbers (Page 8), *infinity* does not exist in the real world,

EXAMPLE 0.65. We may say that the number of molecules in a spoonful of milk is infinite, but of course it's just that the number of molecules is too large for us to count under a microscope.

So, in accordance with the real world, we will employ nearby numbers, that is, in this case, numbers **near** ∞, that is *infinite* numbers,

EXAMPLE 0.66. $-12\,729\,000\,307$ and $+647\,809\,010\,374$ are both near ∞

iii. As we saw in **?? ??** - **??** (**??**), measured numbers will always differ from a given number x_0 by some error

EXAMPLE 0.67. I can give you 3 apples but I cannot give you a 3 foot long stick as it will always be a bit too long or a bit too short.

So, in accordance with the real world, we will employ nearby numbers that is, in this case, numbers **near** x_0, that is numbers that *differ* from x_0 by only *infinitesimal* numbers.

EXAMPLE 0.68. $-87.36 \oplus -0.000.032 = -87.360\,032$ and $-87.36 \oplus +0.000.164 = -87.359\,836$ are both near -87.36

Actually, it is completely standard to speak of a

DEFINITION 0.11 Neighborhood of a point:

- A neighborhood of 0 consists of the numbers *near* 0.
- A neighborhood of ∞ consists of the numbers *near* ∞,
- A neighborhood of x_0 consists of the numbers *near* x_0.

(https://en.wikipedia.org/wiki/Neighbourhood_(mathematics))

And, in fact, we will often speak of **thickening** a given point, that is we will be looking at that point as just the **center** of a neighborhood of that point.

indeterminate number
nearby variable

3. Evaluation *near* a given point. In order to evaluate a global expression *near* a given point, we will evaluate the global expression *At* an **indeterminate *number*** *near* the given point. In other words:

- Instead of declaring 0, we will declare the infinitesimal variable h,
- Instead of declaring ∞, we will declare the infinite variable L,
- Instead of declaring x_0, we will declare:

> **DEFINITION 0.12** The **nearby variable** $\boldsymbol{x_0 \oplus h}$ is the (standard) symbols for numbers near x_0
>
> "Nearby" because, since h is near 0, $x_0 \oplus h$ will be near x_0

Why "circa"? Because nearby is already used.

In other words, we will employ PROCEDURE 0.1 - Get an individual expression from a global expression (Page 14) but with an *indeterminate* number instead of a *given* number.

PROCEDURE 0.3 To evaluate a given global expression in terms of x *near* a given point:

i. Declare an indeterminate numbers *near* the given point, that is:

- If the given point is 0, declare the small variable h by writing the declaration $\Big|_{x \leftarrow h}$, read "x *to be* replaced by h", to the right of the global expression:

$$\text{global expression in terms of } x\Big|_{x \leftarrow h}$$

- If the given point is ∞, declare the large variable L by writing the declaration $\Big|_{x \leftarrow L}$, read "x *to be* replaced by L", to the right of the global expression:

$$\text{global expression in terms of } x\Big|_{x \leftarrow L}$$

- If the given point is x_0, declare the local variable $x_0 \oplus h$ by writing the declaration $\Big|_{x \leftarrow x_0 \oplus h}$, read "$x$ *to be* replaced by $x_0 \oplus h$", to the right of the global expression:

$$\text{global expression in terms of } x\Big|_{x \leftarrow x_0 \oplus h}$$

ii. Replace every occurence of x in the global expression in terms of x by the declared variable to get the global expression for numbers near the given point :

- global expression in terms of h for numbers near 0
- global expression in terms of L for numbers near ∞
- global expression in terms of $x_0 \oplus h$ for numbers near x_0

iii. Execute the general expression in terms of the declared variable according to the relevant rules in **??** **??** - **??** (**??**)

In contradistinction with **??** **??** - **??** (**??**), we have:

DEMO 0.3a To evaluate the global expression $\dfrac{x^{+2} \ominus +7}{x \oplus +3}$ *near* $+5$

i. We declare that the numbers are to be near $+5$ by writing the declaration $\Big|_{x \leftarrow +5 \oplus h}$, read "$x$ *to be* replaced by $+5 \oplus h$", to the right of the global expression:

$$\left.\frac{x^{+2} \ominus +7}{x \oplus +3}\right|_{x \leftarrow +5 \oplus h}$$

ii. We replace every occurence of x in the global expression in terms of x by the local variable $+5 \oplus h$ to get the global expression for numbers near $+5$:

$$\frac{\left(+5 \oplus h\right)^{+2} \ominus +7}{+5 \oplus h \oplus +3}$$

iii. We execute the global expression in terms of $+5 \oplus h$:

$$\frac{+25 \oplus +10h \oplus +h^2 \ominus +7}{+5 \oplus +h \oplus +3}$$

$$\frac{+18 \oplus +10h \oplus +h^2}{+8 \oplus +h}$$

Since the division probably won't stop by itself and since where *RBC* wiil stop the division will depend on the information we will want, the last expression just above is not an executed expression.

In contradistinction with **?? ??** - **??** (**??**), we have:

Demo 0.3b To evaluate the global expression $\dfrac{x^{+2} \ominus +7}{x \oplus +3}$ *near* -3

i. We declare that the numbers are to be near -3 by writing the declaration $\Big|_{x \leftarrow -3 \oplus h}$, read "$x$ *to be* replaced by $-3 \oplus h$", to the right of the global expression:

$$\left.\frac{x^{+2} \ominus +7}{x \oplus +3}\right|_{x \leftarrow -3 \oplus h}$$

ii. We replace every occurence of x in the global expression in terms of x by the local variable $-3 \oplus h$ to get the global expression for numbers near -3:

$$\frac{-3 \oplus h^{+2} \ominus +7}{-3 \oplus h \oplus +3}$$

iii. We execute the global expression in terms of $-3 \oplus h$:

$$\frac{+9 \oplus -6h \oplus h^2 \ominus +7}{-3 \oplus +3 \oplus h}$$

$$\frac{+2 \oplus -6h \oplus h^2}{h}$$

$$+2h^{-1} \oplus -6 \oplus h$$

Since the division was by h, the last expression just above is an executed expression.

In contradistinction with **?? ??** - **??** (**??**), we have:

Demo 0.3c To evaluate the global expression $\dfrac{x^{+2} \ominus +9}{x \oplus -3}$ *near* $+3$

i. We declare that the numbers are to be near $+3$ by writing the declaration $\Big|_{x \leftarrow +3 \oplus h}$, read "$x$ *to be* replaced by $+3 \oplus h$", to the right of the global expression:

$$\left.\frac{x^{+2} \ominus +9}{x \oplus -3}\right|_{x \leftarrow +3 \oplus h}$$

ii. We replace every occurence of x in the global expression in terms of x by the local variable $+3 \oplus h$ to get the global expression for numbers near $+3$:

$$\frac{+3 \oplus h^{+2} \ominus +9}{+3 \oplus h \oplus -3}$$

iii. We execute the global expression in terms of $+3 \oplus h$:

$$\frac{+9 \oplus +6h \oplus h^2 \ominus +9}{+3 \oplus -3 \oplus h}$$

$$\frac{+6h \oplus h^2}{h}$$

$$+6 \oplus h$$

Note that, here, the division being by h, we just did it and the expression just above is an executed expression.

And here is how it goes near ∞:

DEMO 0.3d To evaluate the global expression $\dfrac{x^{+2} \ominus +9}{x \oplus -3}$ *near* ∞

i. We declare that the numbers are to be near ∞ by writing the declaration $\Big|_{x \leftarrow L}$, read "x *to be* replaced by L", to the right of the global expression:

$$\left.\frac{x^{+2} \ominus +9}{x \oplus -3}\right|_{x \leftarrow L}$$

ii. We replace every occurence of x in the global expression in terms of x by the local variable L to get the global expression for numbers magnifier

near ∞:

$$\frac{L^{+2} \ominus +9}{L \oplus -3}$$

iii. We execute the global expression in terms of L:

$$\frac{L^2 \ominus +9}{L \ominus -3}$$

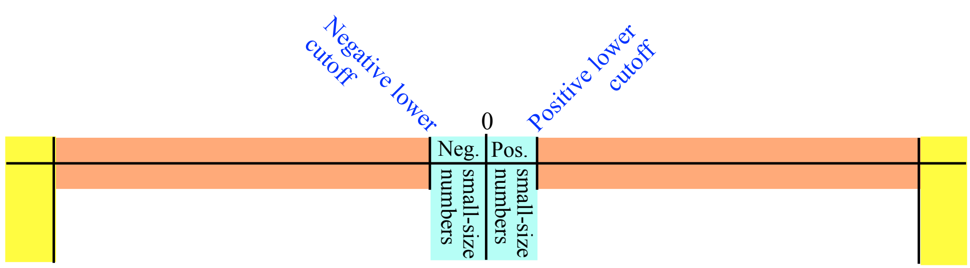
$$\frac{L^2 \ominus [...]}{L \ominus [...]}$$

$$\frac{L^2}{L} \oplus [...]$$

$$L \oplus [...]$$

The last expression just above is the executed expression.

4. Picturing a neighborhood of 0. In ?? ?? - ?? (??), infinitesimal numbers were pictured with

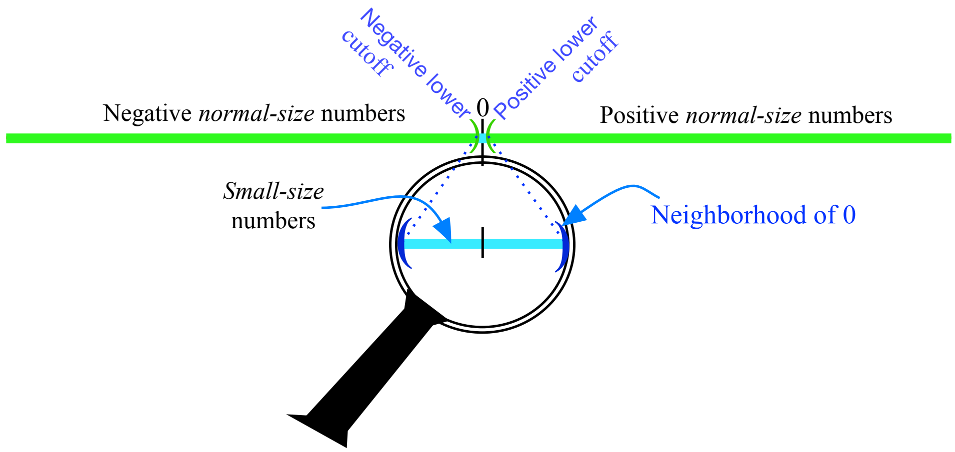


which is *not* really a representation because the three qualitative sizes are represented at different scales. (https://en.wikipedia.org/wiki/Scale_(represent)#Large_scale,_medium_scale,_small_scale).

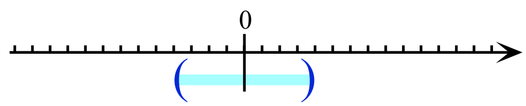
i. On a ruler, at just about any scale (https://en.wikipedia.org/wiki/Scale_(represent)#Large_scale,_medium_scale,_small_scale), the negative lower cutoff for medium-size numbers and the positive lower cutoff for medium-size numbers will both be on top of 0 and we won't be able to see infinitesimal numbers.

So, in order to see a neighborhood of 0, we would need some kind of **magnifier**:

lnumber ine
number line



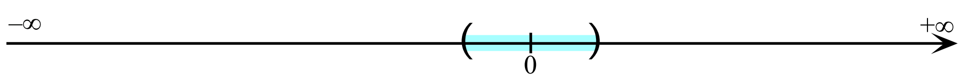
The fact though, that, the neighborhood needs to be representd at a scale larger than the scale of the ruler creates a problem. One way out, of course, would be to draw the neighborhood of 0 just *under* the ruler:



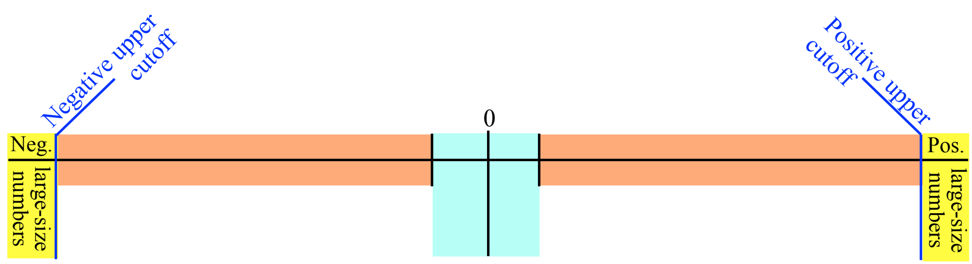
*Can't employ the word **number line** because number lines are tickmarked like rulers.*

ii. So, we cannot employ rulers and we will employ just a **number line**, that is something like a ruler but *without* scale and therefore without tick-marks—not even for 0— but with $-\infty$ and $+\infty$ as end of the line symbols in accordance with AGREEMENT B.1 - 'Number' (without qualifier) (Page 249):

we can draw a neighborhood of 0 as



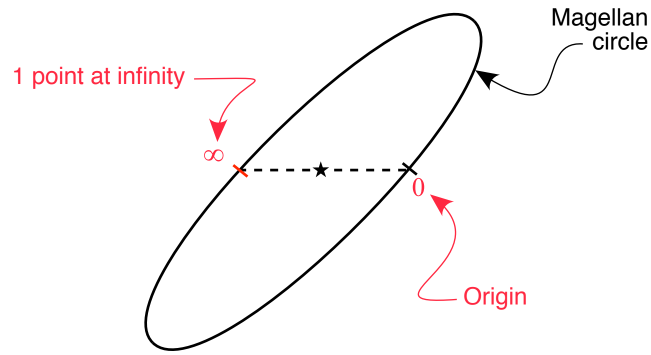
5. Picturing a neighborhood of ∞. In DEFINITION 0.5 - finite number (Page 36) infinite numbers were pictured with



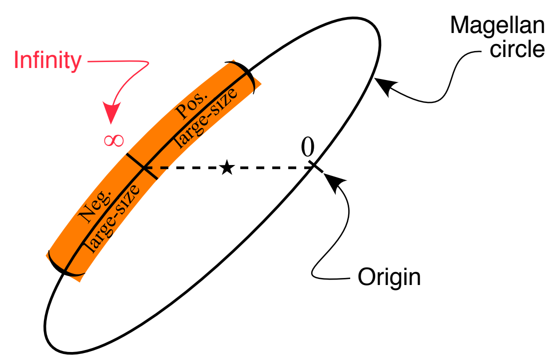
which, again, is *not* a representation because the three qualitative sizes are represented at different scales. (https://en.wikipedia.org/wiki/Scale_(represent)#Large_scale,_medium_scale,_small_scale) compactor

i. On a *quantitative* ruler, at just about any scale, the negative *upper* cutoff for medium-size numbers and the positive *upper* cutoff for medium-size numbers will both be way off the represent so we would need some kind of **compactor**.

ii. In the spirit of one-point compactification, using a Magellan circle

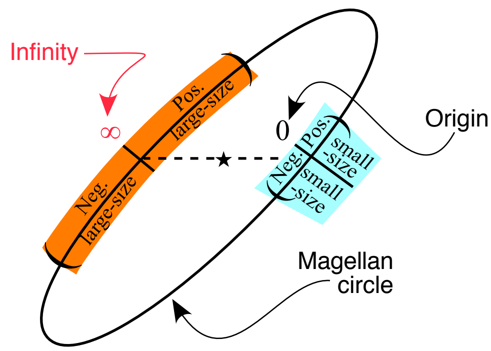


on which infinite numbers are representped as



the advantage is that positive infinite numbers and negative infinite numbers are representped right next to each other the same way as positive infinitesimal numbers and negative infinitesimal numbers:

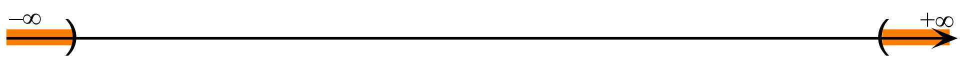
Mercator



Nicely!

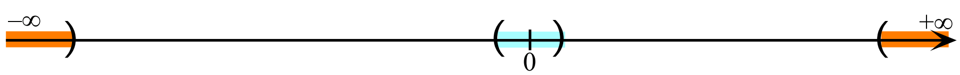
which represents infinite numbers as a neighborhood of ∞ just the way infinitesimal numbers make up a neighborhood of 0.

iii. In the spirit of two-points compactification, we can also represent a neighborhood of ∞, that is infinite numbers, on a line as:



And, after all, 0 is the center of our neighborhood.

Here, the advantage is that we are still facing 0 but the disadvantage is, as opposed to the Magellan represent, that positive infinite numbers and negative infinite numbers are separated from each other, the opposed way of positive infinitesimal numbers and negative infinitesimal numbers which are right next to each other:

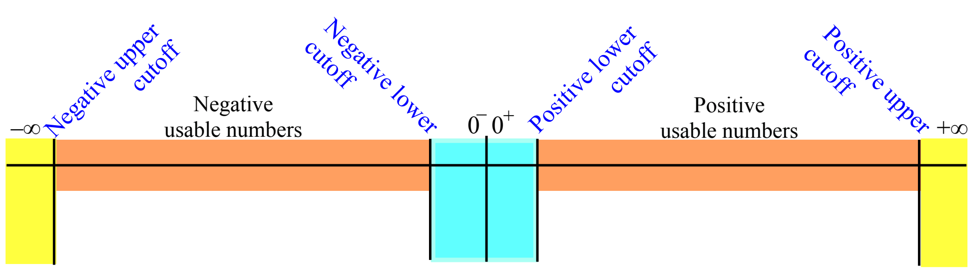


This is often referred to as a **Mercator** represent. (`https://en.wikipedia.org/wiki/Mercator_projection`)

iv.

========End WORK ZONE=======

6. Picturing a neighborhood of x_0. In **??** **??** - **??** (**??**) medium-sized numbers were pictured wirh

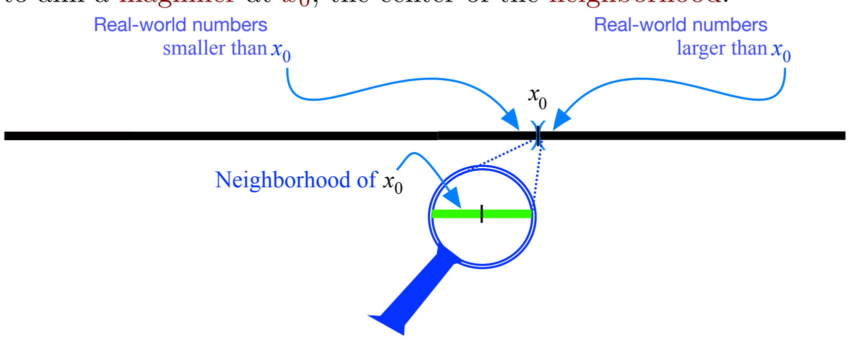


side-neighborhoods
left-neighborhood
right-neighborhood

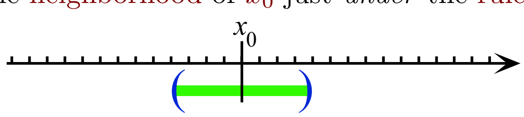
which, again, is *not* a represent because the three qualitative sizes are representd at different scales. (https://en.wikipedia.org/wiki/Scale_(represent)#Large_scale,_medium_scale,_small_scale)

The situation with a neighborhood of x_0 is similar to the situation with a neighborhood of 0:

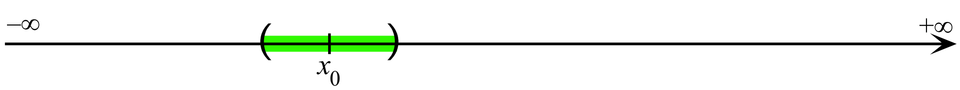
i. On a ruler, at just about any scale (https://en.wikipedia.org/wiki/Scale_(represent)#Large_scale,_medium_scale,_small_scale), the medium-size numbers smaller than x_0 and the medium-size numbers larger than x_0 leave no room between them and we won't be able to see the numbers near x_0
So, in order to see a neighborhood of x_0, that is numbers near x_0, that isnumbers that differ from x_0 by only infinitesimal numbers, we would need to aim a magnifier at x_0, the center of the neighborhood.



Again, the fact that a neighborhood needs to be representd at a scale larger than the scale of the ruler creates a problem. And again, a way out would be to represent the neighborhood of x_0 just *under* the ruler:



ii. But on a *qualitative* ruler we can represent a neighborhood of x_0 as



7. Side-neighborhoods. In order to deal *separately* with each side of a neighborhood we will often have to distinguish the **side-neighborhoods**. Pinning down the **left-neighborhood** from the **right-neighborhood**, though, depends on the nature of the point:

- – A left-neighborhood of 0 consists of the *negative* numbers *near* 0 (*negative* infinitesimal numbers),

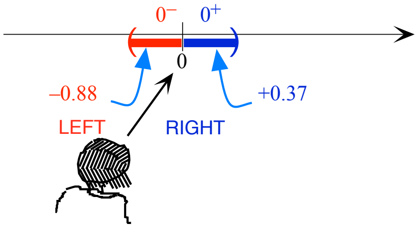
0^+
right
0^-
left
$+\infty$
$-\infty$

– A right-neighborhood of 0 consists of the *positive* numbers *near* 0 (*positive* infinitesimal numbers),

In order to deal *separately* with each side of a neighborhood of 0, we will employ the symbols

- $\mathbf{0^+}$ (namely 0 with a little $+$ up and to the right) which is *standard* expression for positive infinitesimal numbers.
  Positive infinitesimal numbers are **right of** of 0, that is they are to *our right* when *RBC* are facing 0, the center of the neighborhood.
- $\mathbf{0^-}$ (namely 0 with a little $-$ up and to the right) which is *standard* expression for negative infinitesimal numbers.
  Negative infinitesimal numbers are **left** of 0, that is they are to *our left* when *RBC* are facing 0, the center of the neighborhood.

EXAMPLE 0.69. 0^+ refers toinfinitesimalnumbers right of 0 (such as for instance $+0.37$) and 0^- refers to infinitesimal numbers left of 0 (such as for instance -0.88):



So, never forget that

> **CAUTION 0.13** - or - up to the right and *by itself* is *not* an 'exponent' but indicates which *side* of 0.

- – A left-neighborhood of ∞ consists of the *positive* numbers *near* ∞ (*positive*infinite numbers),
  – A right-neighborhood of ∞ consists of the *negative* numbers *near* ∞ (*negative* infinite numbers),

Just as we will often have to refer separately to each side of a neighborhood of 0, we will often have to refer separately to each side of a neighborhood of ∞

BeginWORKzone - BeginWORKzone - BeginWORKzone - BeginWORKzone - BeginWORKzo

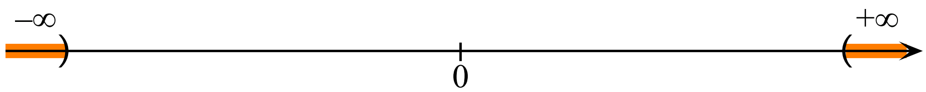
So we will use:

- $+\infty$ as symbol for *positive* infinite numbers,
- $-\infty$ as symbol for *negative* infinite numbers,

even though

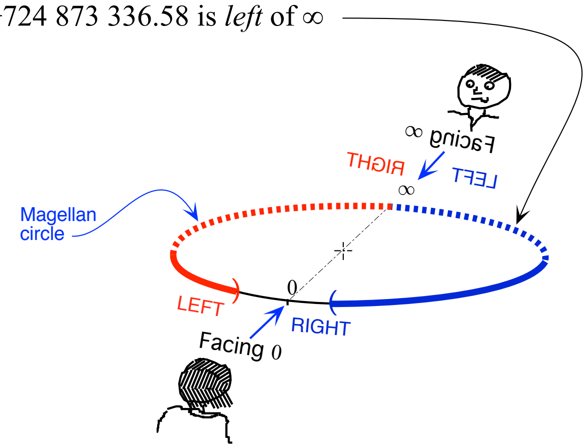
EndWORKzone - EndWORKzone - EndWORKzone - EndWORKzone - EndWORKzone

We will then employ as line:



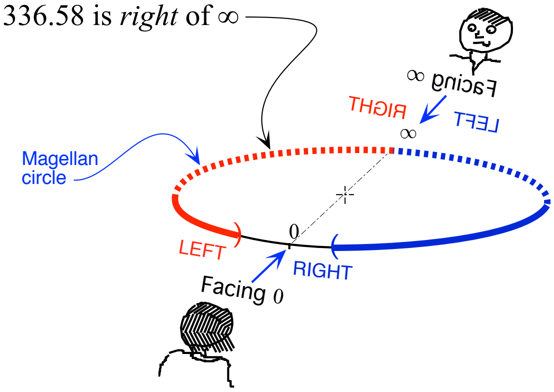
- Keep in mind that it is easy to forget which side is left of ∞ and which side is right of ∞ because it is easy to forget that one must *face* the center of the neighborhood, namely ∞:
  - *Positive* infinite numbers are *left* of ∞ because, to face the center of the neighborhood, we have to imagine ourselves facing ∞, and then *positive* numbers will be to *our* left.

EXAMPLE 0.70. +724 873 336.58 is *left* of ∞



  - *negative* infinite numbers are *right* of ∞ because, to face the center of the neighborhood, we have to imagine ourselves facing ∞, then *negative* numbers would be to *our* right.

EXAMPLE 0.71. –724 873 336.58 is *right* of ∞



- – A left-neighborhood of x_0 consists of the numbers *near* x_0 that are smaller than x_0,(medium-size numbers that differ from x_0 by only infinitesimal numbers).
  – A right-neighborhood of x_0 consists of the numbers *near* x_0 that are larger than x_0,

8. Interplay between 0 and ∞. As already mentioned in Section 4 - Expressions And Values (Page 12), both Expressions And Values have intrigued people for a long time:

i. While, as mentioned in Section 4 - Expressions And Values (Page 12), both 0 and ∞ are literally without meaning, both 0 and ∞ are absolutely and completely indispensable.

EXAMPLE 0.72. When we have eaten three apples out of five apples, we indicate that there are two apples left by writng:

$$5 \text{ apples} - 3 \text{ apples} = 2 \text{ apples}$$

But when we have eaten three apples out of three apples, how do we indicate that there is none left?

$$3 \text{ apples} - 3 \text{ apples} = ? \text{ apples}$$

EXAMPLE 0.73. When we count "eight, nine, ten, eleven" we employ a rhythm as indicated by the commas, say:

eight $1sec$ nine $1sec$ ten $1sec$ eleven

And in fact, when we start counting *with* "eight", we think we are counting *from* "seven" and precede "eight" with the same silence:

$1sec$ eight $1sec$ nine $1sec$ ten $1sec$ eleven

But *from* what number are we thinking we are starting *from* when we start counting *with* "one" and precede "one" by the same silence?

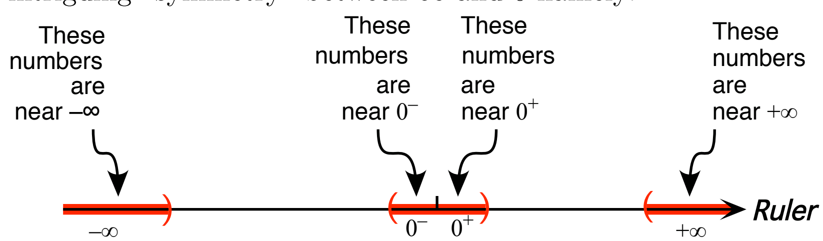
$1sec$ one $1sec$ two $1sec$ three $1sec$ fout

Example 0.74. When we get impatient and want to stop counting, we probably end the counting with "etc"

Example 0.75. When a number is so large that we cannot even begin to imagine it, we often employ the word "infinite".

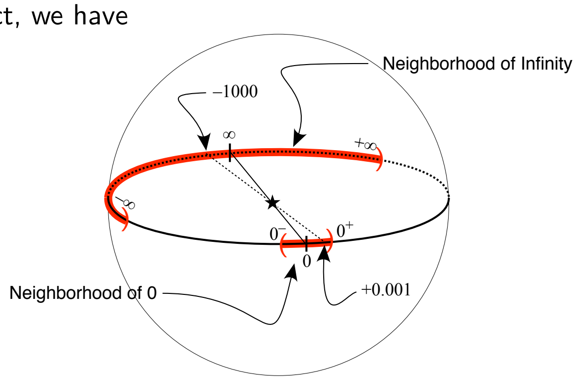
As in "The number of people who want to teach you is infinite."

ii. Even though, as an input, 0 is usually not particularly important, there is an intriguing "symmetry" between ∞ and 0 namely:



More precisley, *small numbers* are some sort of inverted image of *large numbers* since the *reciprocal* of a *large number* is a *small number* and vice versa.

Example 0.76. The opposite of the reciprocal of -0.001 is $+1\,000$. In a Magellan aspect, we have



iii. Moreover, since by **?? ??** - **??** (**??**), infinitesimal numbers are near 0 and infinite numbers are near ∞, Theorem 0.6 - Reciprocity of qualitative sizes (Page 44) can be restated as

Theorem 0.6 (Restated) Reciprocity of qualitative sizes

- The reciprocal of a number near ∞ is a number near 0,
- The reciprocal of a number near 0 is a number near ∞.

It then seems somewhat artificial, even though **?? ?? - ??** (**??**) and CAUTION 0.2 - No other number (Page 5), not to extend the reciprocity of numbers near 0 (infinitesimal numbers) and numbers near ∞ (infinite numbers) to a reciprocity of 0 and ∞ themselves. So,

AGREEMENT 0.6 Since we will *not* compute with ∞, this will only be a shorthand for THEOREM (Restated) 0.6 - Reciprocity of qualitative sizes (Page 59).

But what an extremely convenient shorthand!

Part I

Functions Given By Data

Everything **Connects** To Everything Else.[17]

Leonardo da Vinci

connect

Chapter 1

Relations Given By Data

The truth of the above quote from Da Vinci can be seen everywhere.

EXAMPLE 1.1. Everything sits on something else: people sit on chairs that sit on floors that sit on joists that sit on walls that sit on . . .

All people are six or fewer social connections away from each other.[a]

[a]https://en.wikipedia.org/wiki/Six_degrees_of_separation

And in fact, Da Vinci's statement is at the very heart of all SCIENCES: For a sentence to say something useful about something, we usually must look at that thing in connection to other things.

Even if we *can't always* see, *let alone* understand, *the connections.*

EXAMPLE 1.2. We might say that someone's income tax was $2 270 but, *by itself*, that wouldn't be saying much because

- $2 270 of income tax was a lot more money in Year 1913 —the year income tax was first established, than, say, in Year 2023 . So, for saying that someone's income tax is $2 270 to be useful, we would have to have some relation relating years with Income Tax,

Similarly, because

[17]https://medium.com/@nikitavoloboev/everything-connects-to-everything-else-c6a2d96a809d According to https://quoteinvestigator.com/2022/03/31/connected/ however, the earliest *published* version is from Gotthold Ephraim Lessing in 1769.

Of course, what we would really want is income tax in relation with both year and income

relation
ordered pair
2-tuple
relation
parenthesis
(
)
pair

- $2 270 of income tax is a lot more money for the rest of us than for billionaires, for saying that someone's income tax is $2 270 to be useful, we would have to have some relation relating Incomes with Income Tax.

1 Relations Given By Data-sets

The mathematical concept underlying Da Vinci's connections is that of a **relation** but there are many kinds of relations and many ways to give a relation.

1. Ordered pairs. That a first item is related to a second item in a particular way does not guarantee that the second item will be related to the first item in the same way.

Example 1.3. While "Beth is the sister of Jill" guarantees that "Jill is the sister of Beth", "Jack likes Jane" does not guarantee that "Jane likes Jack".

An **ordered pair** of items then is two items *in a given order*.[18]

Language 1.1
Ordered pairs are also called **2-tuples** but *RBC won't* employ that word.

The standard way to write ordered pairs is with the **pair notation** in which the two items are written in the given order, separated by a comma and enclosed between the **parentheses** (and).

Example 1.4. The ordered pair **(**Eiffel Tower , Empire State Building**)** is *not* the same as the ordered pair **(**Empire State Building , Eiffel Tower**)**

Just like a pair of shoes is not the same kind of pair as a pair of socks.

Caution 1.1 In Mathematics:
- An *ordered* pair

is not to be confused with
- A *pair*, which is just a *collection* of two items so that the order in which the two items in a *pair* are given is *irrelevant*.

Nevertheless, since *RBC* will employ *only* ordered pairs:

[18] https://en.wikipedia.org/wiki/Ordered_pair

Agreement 1.1 The adjective "ordered" will *go without saying* and *RBC* will employ the word **pair** as short for ordered pair.

data-set
left-item
collection of left-items
right-item
collection of right-items
related-pair
collection of related-pairs
related-pair notation
angle
⟨
⟩
unrelated-pair

2. Data-sets. The simplest kind of relation occurs in the discrete aspect of the real world and is given by way of a **data-set** consisting of:

- A **collection of left-items**,
- A **collection of right-items**,

Agreement 1.2
To help distinguish left-items from right-items, *RBC* will employ:
– Pink boxes for left-items as in, for instance,
Jill, x, -0.053, x_0, 0 ∞, *small*, *large*,
– Green boxes for right-items as in, for instance,
Jack, y, $+32.14$, y_0, 0 ∞, *small*, *large*,

Like Port and Starboard on a ship. (https://en.wikipedia.org/wiki/Port_and_starboard)

together with

- A **collection of related-pairs**, that is a collection of (ordered) pairs in which:
  i. The first item is a left-item
  ii. The second item is a right-item

and

iii. The left-item is related—Da Vinci would have said "connected"—to the right-item which will be indicated with the **related-pair notation** in which the **angles** ⟨ and ⟩ replace the parentheses (and) so that $\Big\langle$ left-item, right-item $\Big\rangle$ will say that left-item is related to right-item.

Of course there will also be **unrelated-pairs**, that is (ordered) pairs in which:
i. The first item *is* a left-item,
ii. The second item *is* a right-item

but

iii. The left-item is *not* related to the right-item so that *RBC* cannot employ the related-pair notation and can only write $\Big($ left-item, right-item $\Big)$.

source
target
graph

AGREEMENT 1.3 *RBC* will employ the word pair when we don't know or don't care whether the pair is a related-pairs or a unrelated-pair.

LANGUAGE 1.2 For the sake of immediate transparency, *RBC* will *not* employ the following standard words:

- **Source** for the collection of left-items,
- **Target** for the collection of right-items,
- **Graph** for the collection of related-pairs—but this to keep the word graph for the *picture* of the collection of related-pairs.

Also,

- *RBC* will *not* employ the word **data** just by itself because the word data just by itself is just (standard) jargon for *given* information[a].

[a] https://en.wikipedia.org/wiki/Data

CAUTION 1.2 Readers curious about how relations are dealt with in other books should always make sure what calculus words are being employed for collection of left-items and collection of right-items because calculus words other than source and target *can* be employed[a].

[a] https://en.wikipedia.org/wiki/Relation_(mathematics)

An interesting consequence of Da Vinci's statement is that, in fact, any given item is "known" only by what is already known of the items that the given item is connected/related to.

EXAMPLE 1.5. Sayings about the idea that items are known by what is known of the items they are connected to are found in many cultures[a]:

What a shame though! Such
And the reader confused.
Which should you start with, then?

ram
am
ow

You tell me the company you keep, I will then tell you what you are	(Dutch)
You tell me who's your friend, I will then tell you who you are	(Russian)
You tell me your company, I will then tell you who you are	(Irish)
You tell me what you are eager to buy, I will then tell you what you are	(Mexican)
You tell me with whom you go, I will then tell you what you do	(English)
You tell me who your father is, I will then tell you who you are	(Philippine)
You tell me what you eat, I will then tell you what you are	(French)

[a] https://www.linkedin.com/pulse/show-me-your-friends-ill-tell-you-who-really-jan-johnston-osburn

3. Arrow diagrams, list, tables. The reason we began with relations given by data-set even though Discrete Mathematics is *not* part of Calculus, is that relations given by data-sets are the easiest to give.

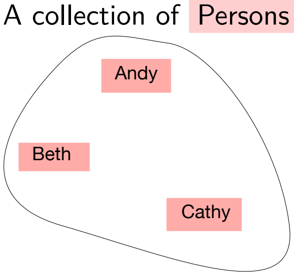
In other words, like collections of whole numbers.

i. Arrow diagrams. The most immediately transparent way to give a data-set is by way of an **arrow diagram**, in which:

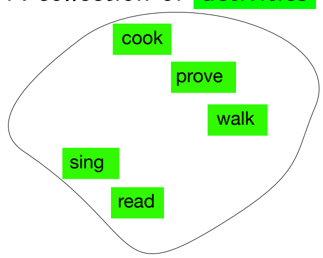
a. The collection of left-items and the collection of right-items are both given by way of **Venn diagrams**[19]

Example 1.6. Venn diagrams for:

A collection of Persons:



A collection of activities

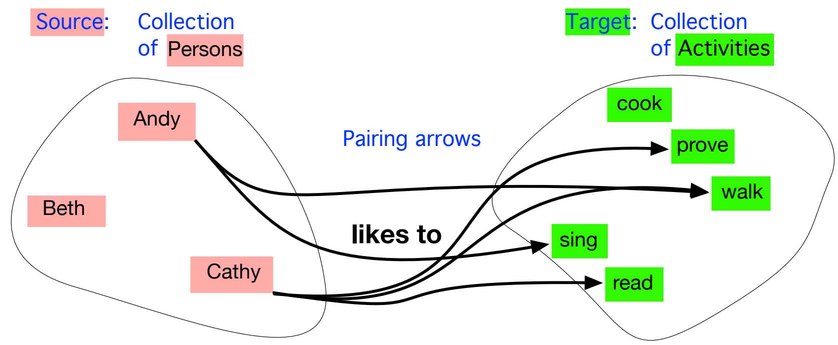


b. The collection of related-pairs is given by way of **pairing-arrows**.

Example 1.6. (Continued) The two collection of items could be connected by pairing-arrows into, for instance, the following arrow diagram:

[19] https://en.wikipedia.org/wiki/Venn_diagram

Couldn't resist arrows going from Source to Target!



In plain English: Cathy likes to prove but not to sing.

which says, for instance, that Cathy likes to prove and that Cathy does *not* like to sing.

Moreover:

- Since Jack is *not* in the collection of Persons, the pair (Jack, prove) is just a pair,
- Since swimming is *not* in the collection of activities, the pair (Beth, swimming) is just a pair,
- Since Cathy does *not* like to cook, the pair (Cathy, cook) is an unrelated-pair,
- Since Andy likes to walk, the pair (Andy, walk) *is* a related-pair which we can therefore write $\langle$ Andy, walk $\rangle$.

While arrow diagrams are very transparent, a limitation of arrow diagrams is that there can only be a very few items in the collections.

ii. Lists A perhaps less transparent, but certainly much more efficient, way to give data-sets than to employ arrow diagrams is just to write the collections as lists:

a. The collection of left-items and the collection of right-items by way of two lists of items.

Example 1.6. (Continued) List of items:

List of Persons:	List of activities:
Andy, Beth, Cathy.	walk, sing, cook, prove, read.

b. The collection of related-pairs by way of a list of related-pairs.

table
row
column
list table

EXAMPLE 1.6. (Continued) List of related-pairs:
⟨Andy, walk⟩, ⟨Andy, sing⟩, ⟨Cathy, walk⟩,
⟨Cathy, read⟩, ⟨Cathy, prove⟩

taken from from the list of *all* possible pairs

(Andy, cook) (Andy, prove) (Andy, walk) (Andy, sing) (Andy, read)
(Beth, cook) (Beth, prove) (Beth, walk) (Beth, sing) (Beth, read)
(Cathy, cook) (Cathy, prove) (Cathy, walk) (Cathy, sing) (Cathy, read)

Lists are clear and allow for quite a few items in the collections—but still not very many,

iii. Tables Using lists to display data-sets, though, can be tedious unless in the shape of **tables** where the lists are in **rows** and **columns** in a way that makes the (left-item, right-item) pairs easy to see. (https://en.wikipedia.org/wiki/Table_(information))

Among different kinds of tables, there are:

- **List tables** in which the collection of left-items is listed in the lefthand column and next to each left-item the related right-item(s), if any, are listed horizontally.

Because the height of a page is larger than the width, some prefer it backwards: left-items listed horizontally *and, under each left-item, the related right-item(s), if any, listed vertically.*

EXAMPLE 1.6. (Continued) Given by list table:

Persons	activities, if any, that Persons, if any, like		
Andy	walk	sing	
Beth			
Cathy	read	walk	prove
	cook		

where, for instance, the following part of the table

Persons	activities, if any, that Persons, if any, like	
Cathy		walk

Cartesian table

says that the sentence ' Cathy likes to walk ' is TRUE that is, in other words, that the pair (Cathy , walk) is a related-pair which therefore we *can* write ⟨ Cathy , walk ⟩.

Just a bit less obvious to read, though.

- **Cartesian tables** which are much more systematic than list tables:

– All the left-items are listed in a vertical column on the left,
– All the right-items are listed in a horizontal row on top,
– For each (left-item , right-item) the word TRUE or FALSE at the intersection of the horizontal row of the left-item and the vertical column of the right-item indicates whether the sentence " left-item is related to right-item " is TRUE or FALSE, that is whether the pair (left-item , right-item) is a related-pair which we can then write ⟨ left-item , right-item ⟩ or an unrelated-pair which we can only write (left-item , right-item).

EXAMPLE 1.6. (Continued) Given by Cartesian table:

likes to	walk	sing	read	prove	cook
Andy	TRUE	TRUE	FALSE	FALSE	FALSE
Beth	FALSE	FALSE	FALSE	FALSE	FALSE
Cathy	TRUE	FALSE	TRUE	TRUE	FALSE

where, for instance, in the following part of the table

likes to	prove
Cathy	TRUE

the word 'TRUE' says that the sentence ' Cathy likes to prove ' is TRUE that is, in other words, that the pair (Cathy , prove) is a related-pair which therefore we *can* write ⟨ Cathy , prove ⟩.

On the other hand, for instance, in the following part of the table

likes to	sing
Cathy	FALSE

relation problem
forward relation problem

the word 'FALSE' says that the sentence " Cathy likes to sing " is FALSE that is, in other words, that the pair (Cathy , sing) is an unrelated-pair which therefore we *cannot* write with angles but only with parentheses.

4. Forward and backward problems. Given a relation, there are of course many questions we can ask and the way *RBC* will *proceed* to answer these questions will depend on how the relation is *given*:

The simplest question is of course whether a given (left-item , right-item) pair is or is not a related-pair.

This question will in fact turn out to be essential for picturing *relations*.

Example 1.6. (Continued) We may ask:
Does Cathy like to sing ?
Answer: No, because the pair (Cathy , sing) is not a related-pair
Does Cathy like to prove ?
Answer: Yes, because the pair (Cathy , prove) is a related-pair which can be written ⟨Cathy , prove⟩

However, given a relation, the more consequential questions that may be asked are **relation problems**.

A. In a **forward relation problem** the information that is wanted is about a given left-item in terms of the right-item if any that the given left-item is related to:

To which right-item(s) if any, is a given left-item (left-number) related to?

In other words, in a forward problem the information goes from left to right :

The way we read.

A given left-item is related to which right-item(s) if any?

But *how* forward problems will be dealt with will depend on the way the relation (numerical endorelation) is given.

a. List tables make it particularly easy to solve forward problems: look up the given left-item in the left column and you will see the right-item(s) that the given left-item is related to, if any, listed on that row.

EXAMPLE 1.6. (Continued) If we ask for all the activities which Cathy likes, the list table in EXAMPLE 1.6 (Page 67) shows:

Cathy	read	walk	prove

If we ask for all the activities which Beth likes, the list table in EXAMPLE 1.6 (Page 67) shows:

Beth			

And, similarly, the list table in EXAMPLE 1.6 (Page 67) even gives answers to questions such as:
Is there any activity Beth likes? (Answer: No)
Does Cathy like *all* activities? (Answer: No)
Does Andy like *at least one* activity? (Answer: Yes)

b. Cartesian tables are only just a bit harder to employ: look up the given left-item in the left column and the right-items that the given left-item is related to, if any, will be in the columns with the word TRUE.

EXAMPLE 1.6. (Continued) If we ask for all the activities which Cathy likes, the Cartesian table shows:

likes to	walk	sing	read	prove	cook
Cathy	**TRUE**	FALSE	**TRUE**	**TRUE**	FALSE

And if we ask for all the activities which Beth likes, the Cartesian table shows:

likes to	walk	sing	read	prove	cook
Beth	FALSE	FALSE	FALSE	FALSE	FALSE

And, similarly, the Cartesian table even readily answers questions such as:
Is there any activity Beth likes? (Answer: No)
Does Cathy like *all* activities? (Answer: No)
Does Andy like *at least one* activity? (Answer: Yes)

Example 1.7. In ?? (??), we may ask
In 2002, did the business really return $+5000$?

backward problem

Example 1.8. In ?? (??), a *forward* problem might for instance be: What was the profit/loss returned by the business in 1999?
Answer: -2000

Example 1.9. In Example A.27 (Page 239), a *forward* problem might for instance be:

$$75 \text{ cents} \xrightarrow{\mathcal{JOE}} \mathcal{JOE}\left(75 \text{ cents}\right) = y \text{ minutes}$$

that is, how many minutes of parking time will $\mathcal{JOE}$ return for 75 cents ?

Example 1.10. Solving *forward* problem in the real world like figuring how much parking time will three quarters buy you is easy: if nothing else, just put three quarters in the parking meter and see how much parking time you get!

Example 1.11. In ?? (??),
There is no Profit/Loss for Year 2000.

B. In a **backward problem** the information that is wanted is about a given right-item in terms of the left-item(s) (left-number(s)), if any, that is/are related to the given right-item:

Which left-item(s) (left-number(s)), if any is/are related to a given right-item ?

In other words, in a backward problem the information goes from right to left:

Opposite *the way we read.*

A given right-item is related to which left-item(s) (left-number(s)), if any?

But, again, *how* backward problems will be dealt with will depend on the way the relation (numerical endorelation) is given.

a. List tables are fairly unsuited to solving backward problems because you have to hunt for the given right-item in all the rows of the right hand column.

EXAMPLE 1.11. (Continued) If we ask for all the Persons who like to walk, the list table shows:

Persons	activities, if any, that Persons like
Andy	**walk** sing
Beth	
Cathy	read **walk** prove
	cook

If we ask for all all the Persons who like to cook, the list table showa:

Persons	activities, if any, that Persons like
Andy	walk sing
Beth	
Cathy	read walk prove
	cook

And similarly, the list table even answers questions such as:

Is there at least one Person who likes to cook? (Answer: No)
Is there at least one Person who likes to walk? (Answer: Yes)
Do *all* Persons like to walk? (Answer: No)

b. Cartesian tables, on the other hand, make it just as easy to solve backward problems as to solve forward problems: look up the given right-item in the top row and the left-item(s) that are related to the given right-item, if any, will be in the rows with the word TRUE.

EXAMPLE 1.11. (Continued) If we ask for all the Persons who like to walk, the Cartesian table shows:

likes to	walk				
Andy	**TRUE**				
Beth	FALSE				
Cathy	**TRUE**				

If we ask for all the Persons who like to cook, the Cartesian table shows:

likes to					cook
Andy					FALSE
Beth					FALSE
Cathy					FALSE

And, similarly, the Cartesian table even answers questions such as:

Is there at least one Person who likes to cook? (Answer: No)
Is there at least one Person who likes to walk? (Answer: Yes)
Do *all* Persons like to walk? (Answer: No)

EXAMPLE 1.12. In ?? (??), a *backward* problem might for instance be: In what year(s) (if any) did the business return $+5\,000$?
Answer: 1998, 2001, 2005.

EXAMPLE 1.13. In EXAMPLE A.27 (Page 239), a *reverse* problem might for instance be:

$$x \text{ cents} \xrightarrow{\mathcal{JOE}} \mathcal{JOE}(x \text{ cents}) = 50 \text{ minutes}$$

that is, how many cents should we input for $\mathcal{JOE}$ to return 50 minutes parking time?

Of course, backward problems *do not have to* have a solution any more than forward problems do .

EXAMPLE 1.14. In ?? (??),
There is no Year for which the Profit/Loss is 6000.

As might perhaps have been expected, *backward* problems are harder to solve—it's, as will be seen, what 'solving equations' is all about, but also what matters most in the real world.

EXAMPLE 1.15. What we usually need to solve in the real world is, for instance, given that we must park for 45 minutes parking time, how many quarters we need to put in the parking meter.

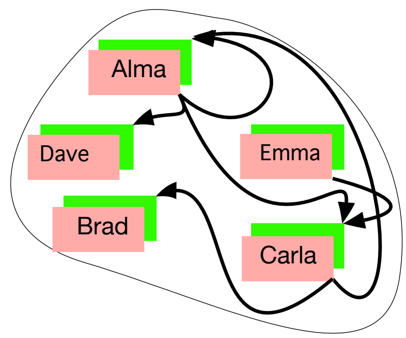
5. Endorelations. There is no reason why the collection of left-items and the collection of right-items cannot be one and the same. and when the collection of left-items and the collection of right-items are one and

Just for the sake of precision! endorelation

the same collection of items, the relation is called an **endorelation**[20].

EXAMPLE 1.16.

Arrow diagram:



List table:

Persons	Persons, if any, whom Persons like		
Alma	Alma	Carla	Dave
Brad			
Carla	Alma	Brad	
Dave			
Emma	Carla		
	Emma		

Cartesian table:

likes	Alma	Brad	Carla	Dave	Emma
Alma	TRUE	FALSE	TRUE	TRUE	FALSE
Brad	FALSE	FALSE	FALSE	FALSE	FALSE
Carla	TRUE	TRUE	FALSE	FALSE	FALSE
Dave	FALSE	FALSE	FALSE	FALSE	FALSE
Emma	FALSE	FALSE	TRUE	FALSE	FALSE

6. Numerical relations In *RBC*, both left-items and right-items will be signed decimal numbers and so:

[20] https://en.wikipedia.org/wiki/Homogeneous_relation

collection of numbers
left-number
right-number
collection of left-numbers
collection of right-numbers
numerical endorelation

LANGUAGE FOR *NUMERICAL RELATIONS* (I) Instead of:	*RBC* will employ:
left-item	**left-number**
right-item	**right-number**
collection of left-items	**collection of left-numbers**
collection of right-items	**collection of right-numbers**

Then, keeping in mind that x_0 and y_0 are symbols for generic given numbers, that is numbers that *you*, the reader, will give, Numbers and infinity (Subsection 7.2, Page 26), *RBC* will employ x_0 as generic given left-number and y_0 as generic given right-number.

However:

CAUTION 1.3 A numerical relation need *not* be an endorelation because even though the left-numbers and the right-numbers both have to be signed decimal numbers, the *collection* of left-numbers need not be the same as the *collection* of right-numbers.

EXAMPLE 1.17. The relation in which the left-items are -3.78, $+1.07$, $+17.0$ and the right-items are -22, $+34$ is a numerical relation but *cannot* be an endorelation whatever the related pairs are.

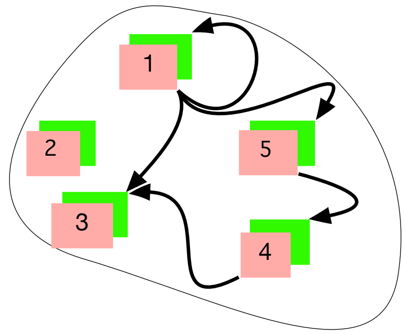
7. Numerical endorelations Often, though, it will be convenient to consider numerical relations in which the collections of left-numbers *is* the same as the collections of right-numbers which *RBC* will thus refer to as **numerical endorelations**.

EXAMPLE 1.17. (Continued) On the other hand, a relation in which the collection of left-numbers and the collection of right-numbers are both the collection of all signed decimal numbers *will* be a numerical endorelation regardless of what the related pairs are.

A numerical endorelation can be given like any relation that is by an arrow diagram, a list, and tables.

EXAMPLE 1.18. A numerical endorelation whose collection of numbers consists of the numbers 1, 2, 3, 4, 5, can be given by an arrow diagram such sparse

as:



or by the corresponding list table:

Left-numbers :	Right-numbers, if any, that the left-numbers are related to:
1	1 3 5
2	
3	
4	3
5	4

However, giving numerical endorelations by data-sets presents a problem in that ,

CAUTION 1.4 Collections of left-numbers are **sparse** that is, there can be only so many left-number in a data-set.

Parentheses

In other words, the gaps between the left-numbers in the collection of left-numbers limit the information we can get.

EXAMPLE 1.19. The data-set consisting of the collection of left-numbers
$\ldots -3, -2, -1, 0, +1, \ldots$
together with the collection of related-pairs
$\ldots \langle -3, -3\rangle$, $\langle -2, -2\rangle$, $\langle -1, -1\rangle$, $\langle 0, 0\rangle$, $\langle +1, +1\rangle$, $\langle +2, +2\rangle$, $\langle +3, +3\rangle$, $\ldots$
gives a numerical endorelation but no information at all about in-between left-numbers

Since from now on all the relations that RBC will consider will be numerical endorelations,

AGREEMENT 1.4 From now on, *RBC* will employ the word relation as a shorthand for numerical endorelation.

basic picture
left-ruler
left-mark
right-ruler
right-mark
pairing-link

2 Relations Given By Data-plots

Even though arrow diagrams are a very natural and very visual way to picture simple relations, more systematic ways will be needed to picture relations.

1. Basic picture. Since rulers can picture numbers with marks, a relation that involves only a few numbers can easily be pictured with a **basic picture**, that is with just:

- a **left-ruler**, that is a ruler on which to **left-mark** left-numbers,
- a **right-ruler**, that is a ruler on which to **right-mark** right-numbers,

—which have the advantage of giving more systematic pictures of collections of numbers than Venn diagrams do—together with

- a collection of **pairing-links** to picture the related-pairs.

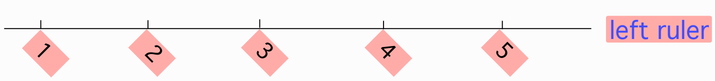
Then,

PROCEDURE 1.1 To get the basic picture of a given numerical endorelation.

i. Draw a left-ruler and left-mark the left-numbers,
ii. Draw a right-ruler and right-mark the right-numbers,
iii. For each related-pair, draw a pairing-link from the left-mark to the right-mark

DEMO 1.1 To get the basic picture of the numerical endorelation given in EXAMPLE 1.19 (Page 78)

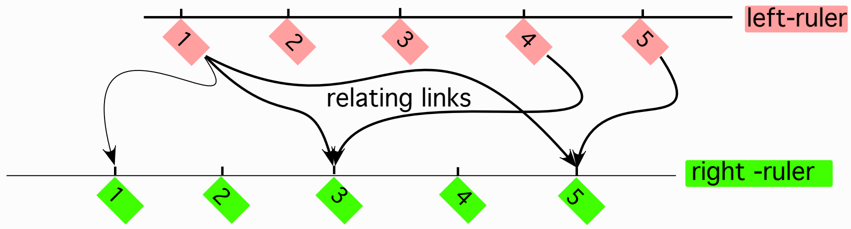
i. We draw a left-ruler and left-mark 1, 2, 3, 4, 5,



ii. We draw a right-ruler and right-mark 1, 2, 3, 4, 5,

Cartesian setup
Descartes
screen

iii. For each related-pair, we then draw a pairing-link from the left-mark to the right-mark



2. Cartesian picture. Even though the left-ruler and the right-ruler in basic pictures provide good pictures of the collection of left-numbers and the collection of right-numbers, pairing-links are not really that different from pairing-*arrows* and since from now on *RBC* will be dealing only with *numbers*, the collections will usually be fairly big so picturing those relations will require a more efficient setup than just basic pictures.

The only issue is that the collection of pairing-links look like a bowl of spaghetti!

The **Cartesian setup**, which is what *RBC* will employ, is due to **René Descartes**[21] who invented ANALYTIC GEOMETRY[22] in order to employ ALGEBRA[23] to solve problems in GEOMETRY[24].

Just like a basic picture, a Cartesian setup involves

- A left-ruler for left-numbers
- A right-ruler for right-numbers

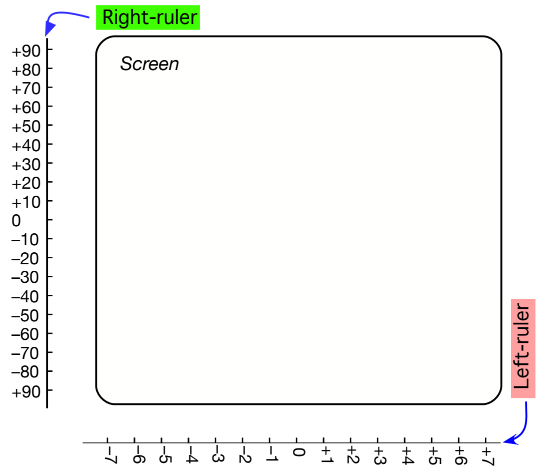
but Descartes' stroke of genius was to employ a rectangular area which *RBC* will call the **screen**, and to draw:

- the left-ruler *horizontally below* the screen
- the right-ruler *vertically left* of the screen

.

EXAMPLE 1.20.

[21] https://en.wikipedia.org/wiki/Ren%C3%A9_Descartes
[22] https://en.wikipedia.org/wiki/Analytic_geometry
[23] https://en.wikipedia.org/wiki/Algebra
[24] https://en.wikipedia.org/wiki/Geometry



pairing-dot
left-number level-line
right-number level-line
relating-dot
non-relating-dot
plain-dot
data point

Yeah, sure enough, Cartesian setups *are upside down from Cartesian* tables *!!!!*

What makes Cartesian setups powerful is that, instead of *related*-pairs being pictured by pairing-links as in the basic picture, in a Cartesian setup *any* pair of numbers can be pictured with just a **pairing-dot**, that is the point at the intersection of

- the **left-number level-line** namely the *vertical* line through the left-mark,

and the

- **right-number level-line** namely the *horizontal* line through the right-mark,

and the pairing-dot can be:

- ▸ a ***relating*-dot** picturing with a *solid* dot ● a *related*-pair,
- ▸ a ***non-relating*-dot** picturing with a *hollow* dot ○ an unrelated-pair,

*Dots relat*ing *left-marks to right-marks to picture relat*ed *pairs of numbers!*

but also just

- ▸ a ***plain*-dot** picturing with an *ordinary* dot • a pair of numbers which might have nothing to do with the relation at hand or, if it does, either we don't know or don't care whether the pair of numbers is a related-pair or an unrelated-pair.

In fact,

- the part of the left-number level-line from the left-mark to the pairing-dot

followed by

- the part of the right-number level-line from the pairing-dot to the right-mark

can be looked-upon as a pairing-link with an elbow at the pairing-dot.

plot

LANGUAGE 1.3 The word "dot" is *not* standard but, because *RBC* is already utilizing the word "point" with a different meaning, *RBC* can neither employ the word **plot point**, standard in MATHEMATICS, nor the word **data point**, standard in the experimental sciences. Subsection 4.1 - Global expressions (Page 12)

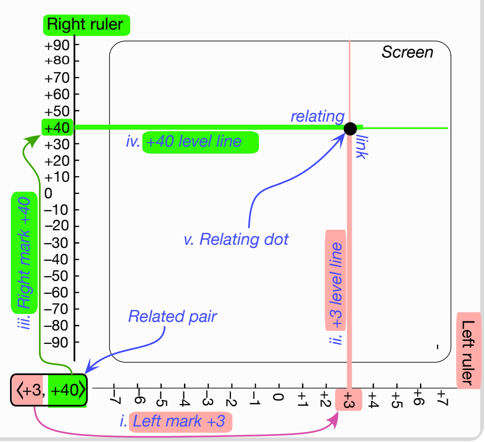
Then,

PROCEDURE 1.2 To **plot** a given pair of numbers (in a Cartesian setup),

i. Left-mark the left-number,
ii. Draw the left-number level-line through the left-mark,
iii. Right-mark the right-number,
iv. Draw the right-number level-line through the right-mark,
v. Mark the intersection of the left-number level-line with the right-number level-line with the appropriate pairing-dot.

DEMO 1.2
Plot the related-pair $\langle +3, +40 \rangle$.

i. We left-mark $+3$,
ii. We draw the $+3$ level-line through the $+3$ mark,
iii. We right-mark $+40$,
iv. We draw the $+40$ level-line through the $+40$ mark,
v. Since $\langle +3, +40 \rangle$ is a related-pair, we mark the intersection of the $+3$ level-line with the $+40$ level-line with a solid dot,

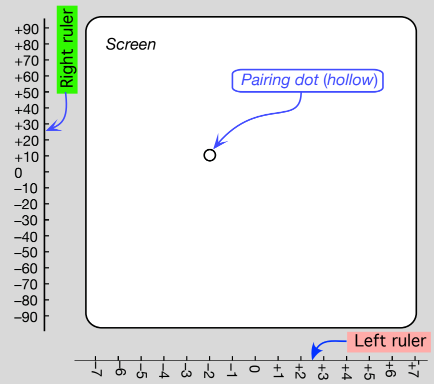


and

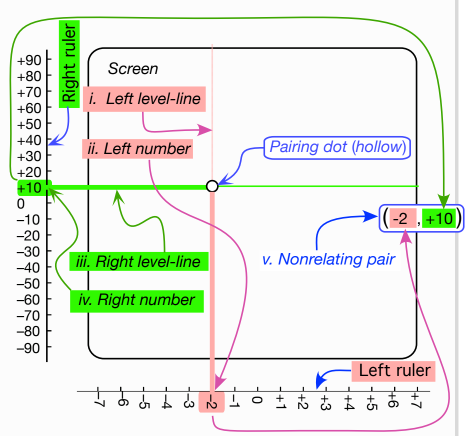
PROCEDURE 1.3 To read the pair of numbers from a given pairing-dot,

i. Draw the left-number level-line through the pairing-dot,
ii. Left-mark the intersection of the left-number level-line with the left-ruler,
iii. Draw the right-number level-line through the pairing-dot,
iv. Right-mark the intersection of the right-number level-line with the right-ruler,
v. Use the appropriate parrentheses for the pair of marked numbers

DEMO 1.3 Get the pair of numbers for the following pairing-dot



i. We draw the left-number level-line through the pairing-dot,
ii. We left-mark the intersection of the left-number level-line with the left-ruler: -2,
iii. We draw the right-number level-line through the pairing-dot,
iv. We right-mark the intersection of the right-number level-line with the right-ruler: $+10$

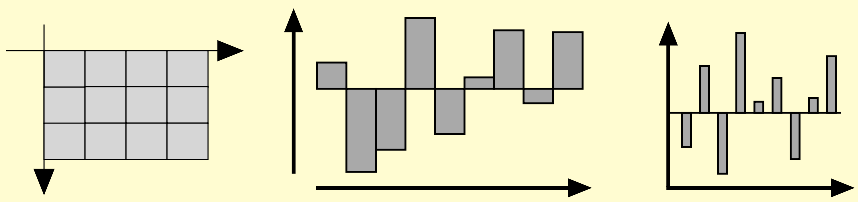


histogram
bar graph

v. Since the given pairing-dot is *hollow*, the pair of marked numbers is *unrelated*: $\left(-2, +10\right)$

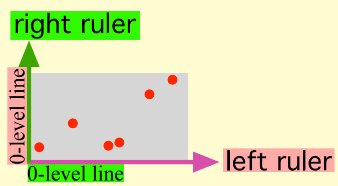
3. Rulers vs. axes.

LANGUAGE 1.4 Keeping the left-ruler and the right-ruler *away* in the offscreen space as *RBC* is doing in the Cartesian setup is standard in the real world:



Cartesian Table **Histogram** [a] **Bar graph**[b]

just as it was for Descartes who, since he did not employ *negative* numbers, could employ the 0 level-line as left-ruler and the 0 level-line as right-ruler which were both out of the way:



Descartes

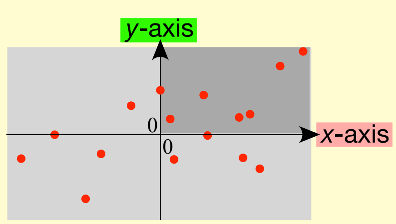
But when *negative numbers* became acceptable, mathematicians continued to employ :

- the 0 level-line as left-ruler

and

- the 0 level-line as right-ruler

even though the left-ruler (called x-axis) and the right-ruler (called y-axis) are now both in the middle of the picture:



x-axis
y-axis
collection of left-marks
collection of right-marks
collection of relating-dots

Modern

[a]See https://en.wikipedia.org/wiki/Histogram
[b]See https://en.wikipedia.org/wiki/Bar_chart

But then:

CAUTION 1.5 Utilizing the x-axis as left-ruler can be confusing because:

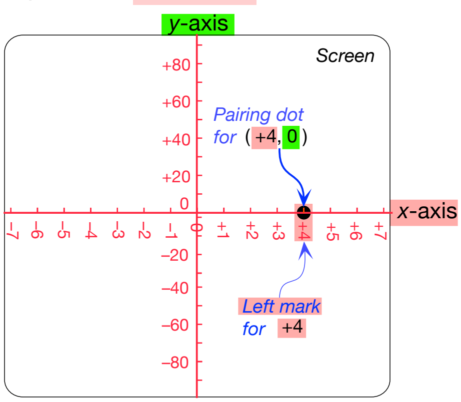
- The pairing-dot for $\left(x_0 , 0 \right)$ will then be on top of the x_0 mark which can make it unclear which is intended,

and so can employing the y-axis as right-ruler because:

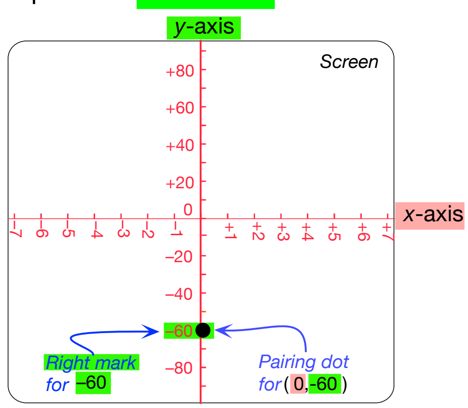
- The pairing-dot for $\left(0 , y_0 \right)$ will then be on top of the y_0 mark which can make it unclear which is intended.

EXAMPLE 1.21. When utilizing the x-axis as left-ruler and the y-axis as right-ruler:

The pairing-dot for $(+4 , 0)$ is on top of the $+4$ mark:



The pairing-dot for $(0 , -50)$ is on top of the -60 mark:



4. Picturing data-sets with data-plots. Since we can mark

- collections of left-numbers as **collections of left-marks**,
- collections of right-numbers as **collections of right-marks**,

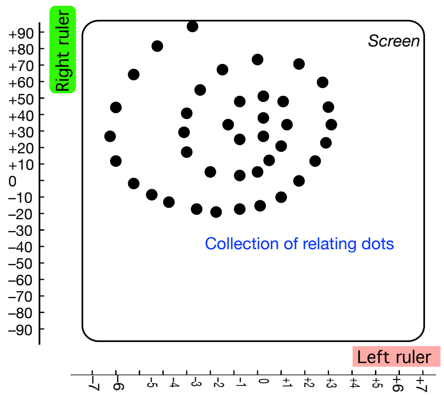
and we can plot

- collections of related-pairs as **collections of relating-dots**,

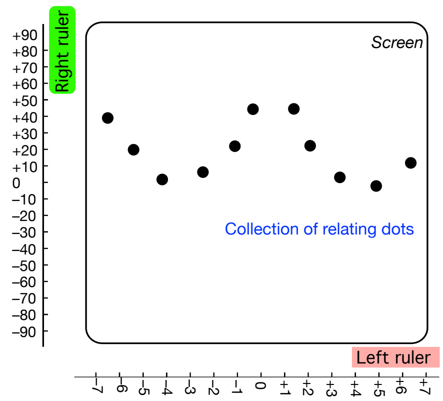
data-plot
quincunx

Cartesian setups allow picturing even large data-sets with **data-plots**, that is with Cartesian setups showing just the collection of relating-dots—but where both left-marks and right-marks are left for the user to get as needed from the relating-dots with PROCEDURE 1.3 - Read a pairing-dot (Page 83)

EXAMPLE 1.22.



EXAMPLE 1.23.

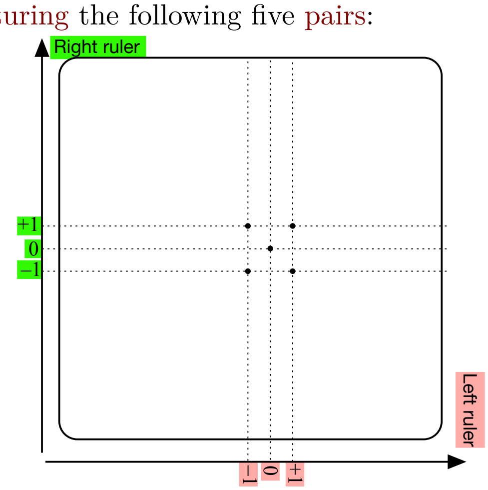


A particular data-plot that *RBC* will employ frequently is the **quincunx**[25], that is the five pairing-dots picturing the following five pairs:

$(-1, +1)$ $(+1, +1)$

$(0, 0)$

$(-1, -1)$ $(+1, -1)$



Note that here the pairing-dots are just plain dots because, *here*, we don't know which of the five (left-number right-number) pairs in the quincunx are *related-pairs* and which are *unrelated-pairs*.

In fact, *which* of the five (left-number right-number) pairs in the quincunx *are* related-pairs will play a central role with power functions.

[25] https://en.wikipedia.org/wiki/Quincunx

In *engineering* and the *experimental sciences*, aside from being given by Cartesian tables, relations are often given by data-plots generated by some machinery[26]. forward problem

However, what can complicate matters is the fact that

Caution 1.6 As a consequence of the Parentheses, data-plots are also sparse.

5. Solving forward problems. To solve a **forward problem** for a relation given left-number when the relation is given by a data-plot, *RBC* will employ

Procedure 1.4 To get the right-number(s) (if any) related to x_0 when the relation is given by a data-plot,

i. Left-mark x_0,

ii. Draw a left-number level-line at the x_0 mark

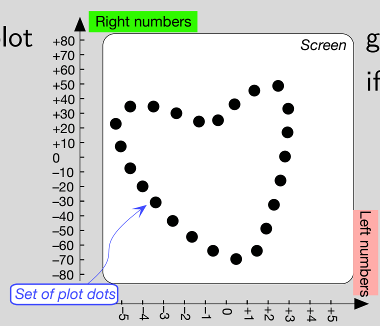
iii. Mark the relating-dot(s) that are on the x_0 level-line, if any—this is where the sparseness of relating-dots comes in,

iv. Draw a right-number level-line through each *marked* relating-dot,

v. Right-mark the right-number(s) related to x_0, if any, on the right-ruler at the right-number level-line(s).

Demo 1.4a

Given the data-plot



get the right-number(s), if any, related to -2.

[26]https://en.wikipedia.org/wiki/Plotter) directly on a Cartesian setup

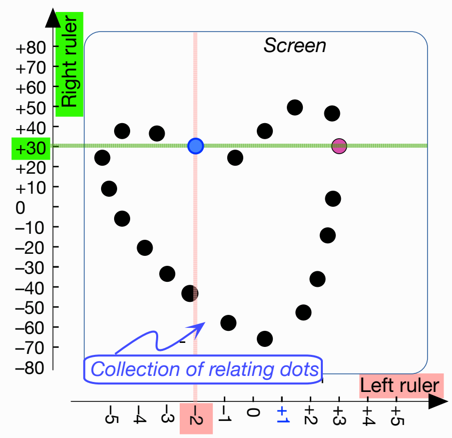
i. We left-mark -2.

ii. We draw the -2 level-line at the -2 mark,

iii. We mark the single relating-dot on the -2 level-line,

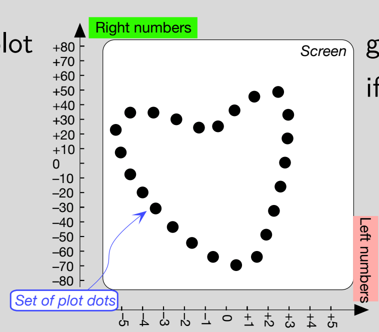
iv. We draw a right-number level-line through the single marked relating-dot,

v. We right-mark on the right-ruler at the single right-number level-line the single right-number related to -2: $+30$.



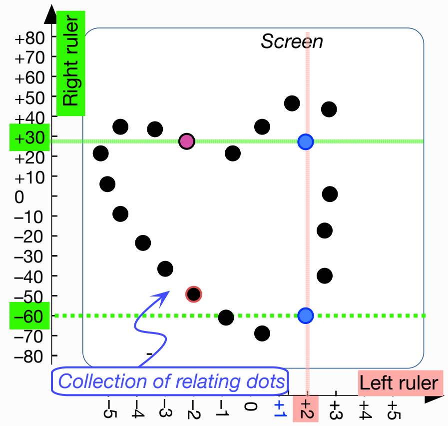
DEMO 1.4b

Given the data-plot



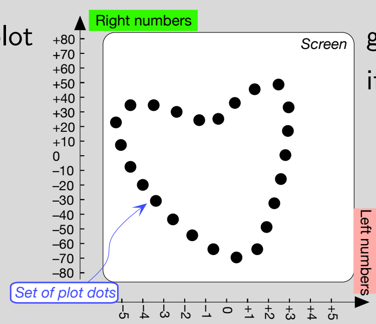
get the right-number(s), if any, related to $+2$.

i. We left-mark $+2$,
ii. We draw the $+2$ level-line at the $+2$ mark,
iii. We mark the two relating-dots on the $+2$ level-line,
iv. We draw the right-number level-line through each of the two marked relating-dots,
v. We right-mark on the right-ruler at the two right-number level-lines the two right-numbers related to $+2$: $+30$ and -60.



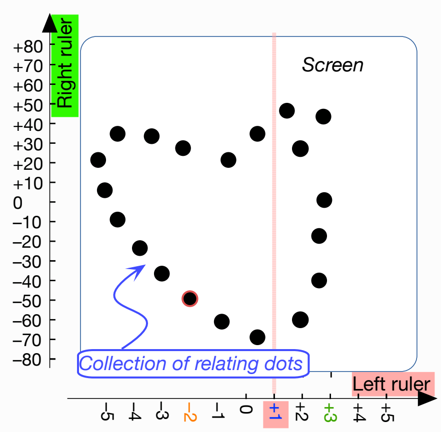
Demo 1.4c

Given the data-plot



get the right-number(s), if any, related to $+1$.

i. We left-mark $+1$.
ii. We draw the $+1$ level-line at the $+1$ mark,
iii. There is *no* relating-dot on the $+1$ level-line to mark—relating-dots are sparse, so there is no right-number related to the given $+1$



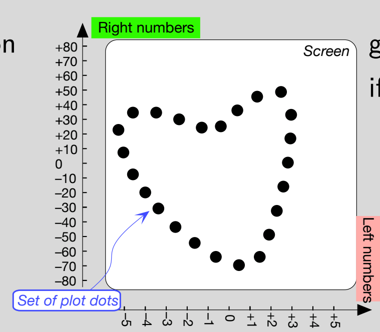
6. Solving backward problems. To solve a backward relation problem for a given right-number when the relation is given by a data-plot, *RBC* will employ

PROCEDURE 1.5 To get the left-number(s) (if any) related to y_0 when the relation is given by a data-plot,

i. Right-mark y_0
ii. Draw a right-number level-line at the y_0 mark
iii. Mark the relating-dot(s) that are on the y_0 level-line, if any—this is where the sparseness of relating-dots comes in,
iv. Draw a left-number level-line through each marked relating-dot,
v. Left-mark the left-number(s) related to y_0, if any, on the left-ruler at the left-number level-line(s)

DEMO 1.5a

Given the relation given by the data-plot



get the left-number(s), if any, related to $+30$.

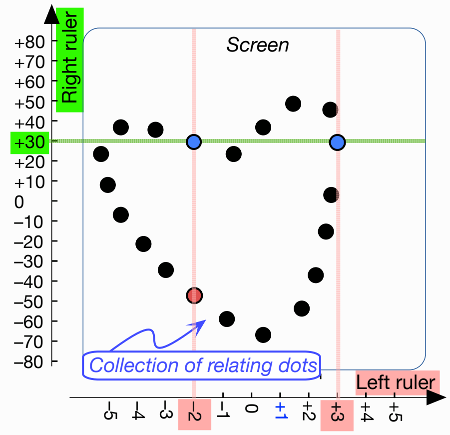
i. We right-mark $+30$.

ii. We draw the $+30$ level-line at the $+30$ mark,

iii. We mark the two relating-dots on the $+30$ level-line,

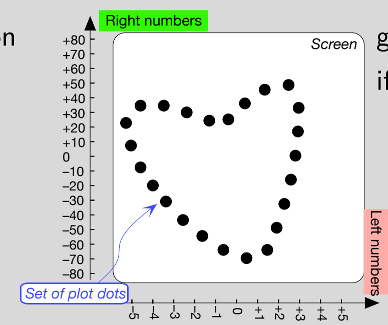
iv. We draw a left-number level-lines through each of the two marked relating-dots,

v. We left-mark on the left-ruler at the two left-number level-lines the two left-numbers related to $+30$: -2, $+3$.



Demo 1.5b

Given the relation given by the data-plot



get the left-number(s), if any, related to -50.

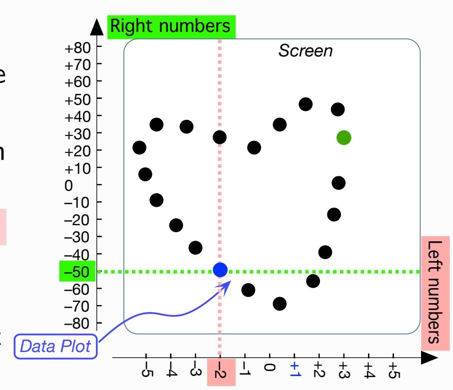
i. We right-mark (the given) -50.

ii. We draw the -50 level-line at the -50 mark,

iii. We mark the single relating-dot on the -50 level-line,

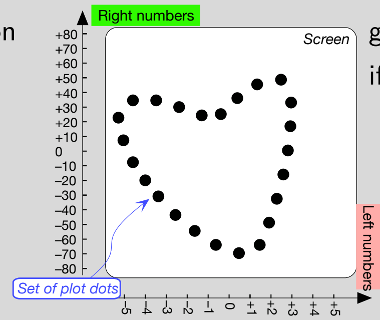
iv. We draw a left-number level-lines through the single marked relating-dot,

v. We left-mark on the left-ruler at the single left-number level-line the single left-number related to -50: -2.

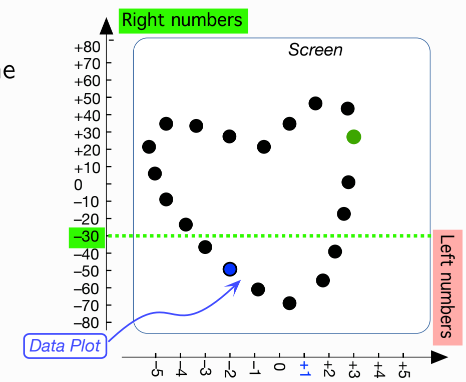


DEMO 1.5c

Given the relation given by the data-plot



get the left-number(s), if any, related to −30.



i. We right-mark −30.

ii. We draw the −30 level-line at the −30 mark,

iii. There is *no* relating-dot on the −30 level-line to mark—relating-dots are sparse, so there is *no* left-number related to the given −30

EXAMPLE 1.24. Given the business in ?? (??) ,

- (1998, +5000) and (2002, −2000) are input-output pairs,
- (1999, +3000) is *not* an input-output pair because the table does *not* pair 1999 with +3 000,
- There is *no* input-output pair involving 2000
- There is no input-output pair involving +3 000

Functions of various kinds are "the central items of investigation" in most fields of modern mathematics.[18]

Michael Spivak[19]

Spivak change

Chapter 2

Functions Given Graphically

Which, of course, is function. See Spivak!

Even though, historically, CALCULUS is short for "calculus of functions"[20], Part I - Functions Given By Data (Page 63) began with *relations* because, as pointed out by Da Vinci, relations are the more immediately universal concept and therefore the background against which functions will make sense.

1 To See Change

CALCULUS is indeed essentially about *how* things **change** and another consequence of Da Vinci's connectivity is that, in order to see how things change, we will *have to* look at these things in relation to *other* things that change differently.

EXAMPLE 2.1. To see that:

- The airplane we are sitting in is moving, we must look out the window.
- The tree out our living room window is growing, we must look at the tree in relation to something like a building.

But then, the fact that a numerical endorelation can relate *one* left-number to *many* right-numbers can make it difficult to see differences between

[18] Calculus, 4th edition. Publish or Perish Press.
[19] https://en.wikipedia.org/wiki/Michael_Spivak
[20] https://royalsocietypublishing.org/doi/10.1098/rstl.1815.0024

function
functional

left-numbers in terms of the right-numbers that these left-numbers are related to.

Example 2.2.

- A *slot machine* can pay for the same number of coins just about any number of coins which makes it quite hard to decide if *this* slot machine is better for gambling than *that* other slot machine.

while

- A *parking meter* can let you park for a number of coins only *one* number of minutes which makes it easy to decide if *this* parking meter is better for parking than *that* other parking meter.

If you are going to read one and only one single Wikipedia article, ***https://en.wikipedia.org/wiki/Function_(mathematics)*** *is absolutely and definitely the one to read—along with* **?? ??** *of course.*

1. To be or not to be functional. Altogether then, even though there are many parts of Mathematics dealing with other kinds of relations, *RBC* will deal with **functions**, that is with numerical endorelations that are **functional** in that they meet

Definition 2.1 The **Functional Requirement**, that is the requirement for a numerical endorelation to be a function, can be stated in different but equivalent ways:

No left-number can be related to *more than one* right-number.

or, in other words,

A left-number can be related to *no more than one* right-number.

or, in other words,

A left-number *cannot* be related to more than one right-number.

or, still in other words,

A left-number can be related to *at most one* right-number.

Example 2.3. In Example 2.2 (Page 94)

- The *slot machine* does *not* meet the **??** because when two Persons play the *same* amount of money in a slot machine, the slot machine can pay *different* amounts of money to the two Persons.
- The *parking meter does* meet the CtitlerefDFN:1-1 because when two Persons put in the *same* amount of money into a parking meter, the parking

meter will always allow the two Persons the *same* number of minutes.

input/output device
I/O device
input-number
input
output-number
output
return

The Functional Requirement makes a *huge* difference between numerical endorelations that *are* functional and numerical endorelations that are *not* functional and this difference is why functions "*are widely used in science, and in most fields of mathematics.*"[21] and why functions will be the *only* numerical endorelations *RBC* will be dealing with.

EXAMPLE 2.4. In contrast with the numerical endorelation in EXAMPLE 1.19 (Page 78), the numerical endorelation given by the table

Left-numbers:	Right-numbers, if any, the left-numbers are related to:
1	2
2	
3	
2	4
5	4

is a function.

With relations, there is often no strong reason for deciding up front which items should be left-items and which items should be right-items.

On the other hand, when dealing in the real world with **input/output devices**[22], **I/O device** for short, that is with what happens to be the exact real world embodiments of functions, there usually is little doubt what should be left-numbers and what should be right-numbers. Hence:

LANGUAGE FOR *FUNCTIONS* (I) Instead of:	*RBC* will employ:
left-number	**input-number** or **input** for short.
right-number.	**output-number** or **output** for short

Moreover, instead of saying that a function relates a left-number to a right-number or that the right-number is related by the function to the left-number, *RBC* will say that the function **returns** the output-number for the input-number—which is also standard.

Some say "the function outputs the output" but RBC will not *because employing the same word as both a noun and a verb can be confusing.*

[21] https://en.wikipedia.org/wiki/Function_(mathematics)
[22] https://en.wikipedia.org/wiki/Input/output

domain

The Functional Requirement can then be restated as:

DEFINITION 2.1 (Restated) Functional Requirement

Given an input-number ,

A function *cannot* return *more than one* output-number .

or, in other words,

A function can return *no more than one* output-number .

or, still in other words,

A function can return *at most one* output-number .

To make perfectly clear what functions *can* do and what functions *cannot* do, here are the answers to the two questions very frequently asked about functions:

i. Does a function *have to* return output-numbers ? In the real world, there is nothing to *force* functions to return an output-number for each and every single input-number .

Of course, you might say that no tax = $0.00 so this may not be a very good example.

EXAMPLE 2.5. Even though incomes below the minimum income are not related to any tax, the income tax meets the ?? and thus " income tax is a function of income ".

In other words, a function *may* or *may not* return an output-number for a given input-number .

EXAMPLE 2.6. The relation in which -23.56 is related to $+101.73$ and no other left-number is related to any right-number meets the functional requirement and *is* thus a function.

But while in the real world, but *outside* of rigourous MATHEMATICS, the Functional Requirement is what is normally used to define a function

LANGUAGE 2.1 In rigourous MATHEMATICS, functions *cannot* be allowed to return *no* output-number because of pathological cases and so, in rigourous MATHEMATICS, the word **domain** is introduced

to refer to the collection of those input-numbers that *are* related to some output-number.
Here again, though—see LANGUAGEE 1.2, the curious reader should keep in mind that in other texts domain may be used in place of source.

However, inasmuch as *RBC* will never get anywhere close to pathological cases:

AGREEMENT 2.1 In *this* text, given an input-number, a function *may* very well return *no* output-number.

ii. *May* a function return *identical* output-numbers for different ? In the real world, there is no reason to *prevent* functions from returning the same output-number for different input-numbers.

EXAMPLE 2.7. A business may be looked upon as the relation given by the table of its profits/losses over the years:

Year	Profit/Loss
1998	$+5\,000$
1999	$-2\,000$
2000	
2001	$+5\,000$
2002	$-2\,000$
2003	$-1\,000$
2004	
2005	$+5\,000$

Even though the same +5 000 occurred in 1998, 2001, and 2005 the business satisfies the **??** and thus "profit/loss is a function of year".

In other words, a function *may* or *may not* return the same output-number for several different input-numbers.

2. Language for functions. Since functions are what CALCULUS deals with, it is of course necesssary for the calculus language to include

f
$\xrightarrow{f}$
$f(x)$
Reverse Polish Notation
RPN
input-output notation
I-O notation
equality notation
arrow notation
arrow-equality notation

language for functions.

i. Naming *generic* functions.

- $\boldsymbol{f}$ as well as $\xrightarrow{f}$ denotes a *generic* function.

together with

- x as a global variable for input-numbers,
- y as a global variable for output-numbers,

Then, the notation $\boldsymbol{f(x)}$, to be read f of x, is the standard notation for the output number, if any, that the function f returns for x.

And just in case you don't have color pens.

CAUTION 2.1 Even though, because *RBC* is employin color boxes, *RBC could* just employ $f\ x$ instead of $f(x)$, *RBC* will still employ parentheses because that's what is done by absolutely everybody.

Even Hewlett-Packard was eventually forced to give up on RPN for its calculators!

LANGUAGE 2.2 Actually, there *is* a parenthesis-free notation called the **Reverse Polish Notation**[a], or **RPN** for short, in which, instead of $f(x)$, the output-number returned by a function f for x is written $x\,f$—*without* parentheses.

The reason *RBC* is *not* employing the RPN is *only* that, unfortunately, about no one in the *mathematical* world employs the RPN and having readers forced to switch sooner or later would be quite uncalled for.

[a] https://en.wikipedia.org/wiki/Reverse_Polish_notation

ii. Functional notations.

While quite standard, $\xrightarrow{f}$ is not (yet?) standard in ... standard CALCULUS books.

Inasmuch as functions are central to CALCULUS, there are many different things to do with functions and so it should not come too much as a surprise that there are several ways to write functions depending on what's to be done:

DEFINITION 2.2 The **input-output notations**[a], **I-O notations** for short, which say that the function f returns the output-number y_0 for the input-number x_0 are:

- For *computing* purposes, the old **equality notation** $f(x_0) = y_0$ has no rival,

alternate arrow notation
send
capital script letters

- For *conceptual* purposes, the modern **arrow notation** $x_0 \xrightarrow{f} y_0$ has no rival either,

  And so, for practically *every* purposes, *RBC* will employ

- The **arrow-equality notation** $x_0 \xrightarrow{f} f(x_0) = y_0$

[a] https://en.wikipedia.org/wiki/Function_(mathematics)#Functional_notation

Language 2.3 Inasmuch as we read from left to right, the fact that the arrow notation

$$x \xrightarrow{f} f(x)$$

starts with x, that is something to be *input* into a function ...yet to be mentioned, is a bit annoying.
So, there is an **alternate arrow notation**

$$f : x \longrightarrow f(x)$$

read

$$f \text{ **sends** } x \text{ to } f(x)$$

which is more satisfying inasmuch as the function is mentioned first.

While *RBC* will *not* employ this *alternate* arrow notation, *RBC will* employ the word send because sending is the mirror image of returning.

iii. Naming *given* functions. To name functions given in **Examples** and **Demos**, as well as "known" functions—to be described later, *RBC* will employ **capital script letters**.

Example 2.8. The function which sends every single input-number to 0 is "known" as the function $\mathcal{ZERO}$.

Example 2.9. Say $\mathcal{JOE}$ is the name of our favorite parking meter. Using the arrow notation, we write

$$x \xrightarrow{\mathcal{JOE}} \mathcal{JOE}(x)$$

and, if $\mathcal{JOE}$ returns 10 minutes parking time when we pay 25 cents, we can write:

zero (of a function)

- For *computational* purposes, $\mathcal{JOE}($ 25 cents $)$ = 10 minutes
- For *conceptual* purposes, 25 cents $\xrightarrow{\mathcal{JOE}}$ 10 minutes
- For *any* purposes, 25 cents $\xrightarrow{\mathcal{JOE}}$ $\mathcal{JOE}\Big($ 25 cents $\Big)$ = 10 minutes
- For *plotting* purposes, $\Big\langle$ 25 cents , 10 minutes $\Big\rangle$

Usually, though, *RBC* will *not* include units in either inputs or outputs.

Example 2.9. (Continued) In the alternate arrow notation, we would write:

$$\mathcal{JOE} : x \longrightarrow \mathcal{JOE}(x)$$

read

$$\mathcal{JOE} \text{ sends } x \text{ to } \mathcal{JOE}(x)$$

and

$$\mathcal{JOE} : \text{25 cents} \longrightarrow \text{10 minutes}$$

read

$$\mathcal{JOE} \text{ sends 25 cents to 10 minutes}$$

which is symmetrical with

$$\mathcal{JOE} \text{ returns 10 minutes for 25 cents}$$

Example 2.10. Given that $\mathcal{JILL}$ returned $+6.75$ for -5.32, we can write

- For *computational* purposes, $\mathcal{JILL}(-5.32) = +6.75$
- For *visual* purposes, $-5.32 \xrightarrow{\mathcal{JILL}} +6.75$
- For *any* purpose, $-5.32 \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(-5.32) = +6.75$
- For *plotting* purposes, $\Big\langle -5.32 , +6.75 \Big\rangle$

And of course, for *plotting* purposes, *RBC* will keep on employing the related-pair notation, $\Big\langle x_0 , y_0 \Big\rangle$, from on to be known as **Input-Output pair notation**, or **I-O pair notation** for short.

3. Zeros and poles. Even though, as discussed in Section 6 - Zero And Infinity (Page 21), 0 and ∞ are not numbers, 0 and ∞ will play a major role in CALCULUS. A bit more precisely:

i. A **zero of a function**[23] f, **zero** of f for short, will be a finite input for which the function returns the output 0.

[23] https://en.wikipedia.org/wiki/Zero_of_a_function

Indeed, 0 will not be especially important as an input but 0 will play a major role as an output and searching for the zero(s), if any, of a given function will in fact be a major backward problem.

ii. Pole of a function. Given a function f, a **pole** of f will be a *finite* input for which the function f returns ∞.

Here again, searching for the pole(s), if any, of a given function will also be a major backward problem.

pole
InputOutput-pair
IO-pair
nonInputOutput -air
nonIO-pair
InputOutput-set
IO-set
InputOutput-pair notation
IO-pair notation
Input-ruler
Output-ruler
Input-mark
Output-mark
collection of Input-marks
collection of Output-marks
Input-level-line
Output-level-line
InputOutput-dot
IO-dot
nonInputOutput-dot
nonIO-dot
InputOutput-plot
IO-plot

2 Functions Given By Input-Output Plots

1. Cartesian language for functions. Because *RBC* is now employing for functions the words input-number and output-number instead of the words left-number and right-number, the language which was introduced in Rulers vs. axes must now be adapted to functions:

LANGUAGE FOR *FUNCTIONS* (II)

Instead of:	*RBC* will employ:
related-pair	**Input Output-pair** or **I O-pair** for short
unrelated-pair	**non Input Output-pair** or **non I O-pair** for short
data-set	**Input Output-set** or **I O-set** for short
left-ruler	**Input-ruler**
right-ruler	**Output-ruler**
left-mark	**Input-mark**
right-mark	**Output-mark**
collection of left-marks	**collection of Input-marks**
collection of right-marks	**collections of Output-mark**
left-number level-line	**input level-line**
right-number level-line	**output level-line**
relating-dot	**Input Output-dot** or **I O-dot** for short
non-relating dot	**non Input Output dot** or **non I O-dot** for short
data-plot	**Input Output-plot** or **I O-plot** for short

and the Functional Requirement can be restated in terms of data-plot:

discrete function

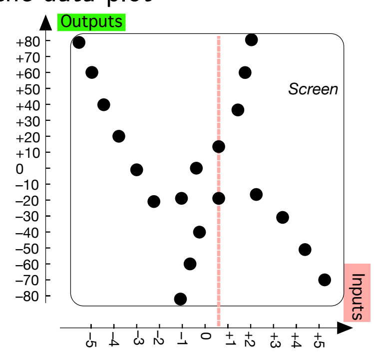
DEFINITION 2.1 (Restated) Functional Requirement
In order for a data-plot to give a function,

No input level-line shall intersect the data-plot more than *once*.

that is, in other words,

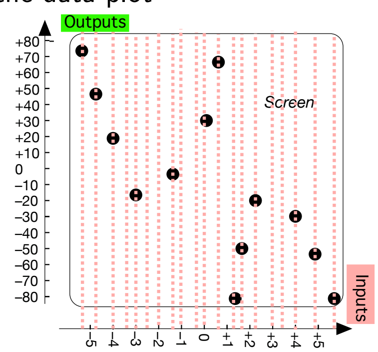
Any input level-line shall intersect the data-plot at most *once*.

EXAMPLE 2.11. Given the data-plot



since there is at least one input level-line that does intersect the data-plot more than *once*, the data-plot *does not* give a *function* and we *cannot* employ the word IO-plot instead of the word data-plot.

EXAMPLE 2.12. Given the data-plot



since *no* input level-line intersects the data-plot more than *once*, the data-plot *does* give a *function* and we can employ the word I O-plot instead of the word data-plot.

By **discrete functions**, *RBC* will mean functions given by an I O-plot.

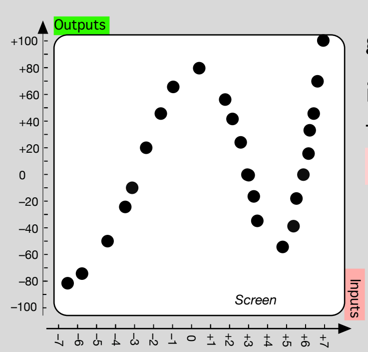
2. Solving forward problems. Solving a forward problem for a locate
function given by an I O-plot, that is **locating** *the* output, if any, that the function returns for a given input, goes of course exactly the same way as solving a forward problem for a relation given by a data-plot—still keeping in mind that Data-plots are sparse:

PROCEDURE 2.1 To get $f(x_0)$ for x_0 when f is given by an I O-plot,

i. Left-mark x_0,
ii. Draw an input level-line through x_0,
iii. Mark the relating dot at the intersection, *if any*, of the input level-line with the I-O plot,
iv. Draw an output level-line through the relating dot (if any),
v. Read $f(x_0)$ where the output level-line intersects the output ruler,
vi. Format the input-output pair according to PROCEDURE 1.5 - right-number for a left-number (data-set (Page 90)

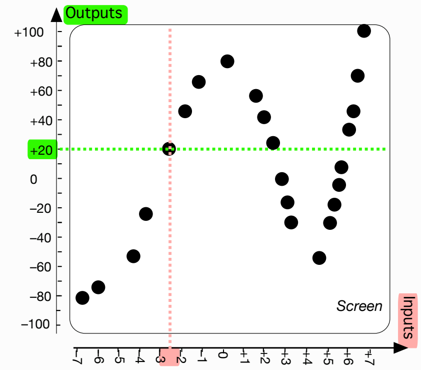
DEMO 2.1a

$\mathcal{JIM}$ being given by the I O-plot



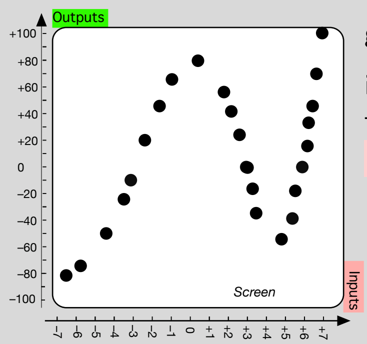
get $\mathcal{JIM}(-2.5)$, that is the output, if any, that $\mathcal{JIM}$ returns for -2.5.

i. We left-mark -2.5,
ii. We draw an input level-line through the -2.5 mark,
iii. We mark the relating dot at the intersection of the -2.5 level-line with the I O-plot
iv. We draw an output level-line through the marked relating dot,
v. We read $\mathcal{JIM}\left(-2.5\right)$ where the output level-line intersects the output ruler: $+20$



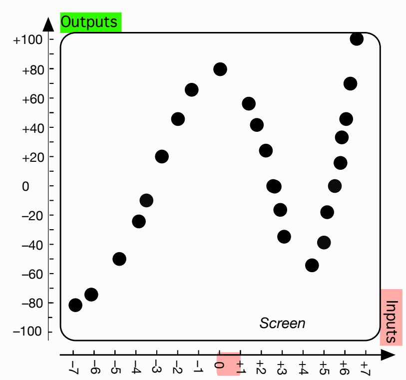
DEMO 2.1b

$\mathcal{JIM}$ being given by the I O-plot



get $\mathcal{JIM}\left(+0.7\right)$, that is the output, if any, that $\mathcal{JIM}$ returns for $+0.7$.

i. We left-mark $+0.7$,
ii. We draw an input level-line through the $+0.7$ mark,
iii. There is no relating dot at the intersection of the $+0.7$ level-line with the I O-plot,
iv. $\mathcal{JIM}$ does *not* return any output for $+0.7$



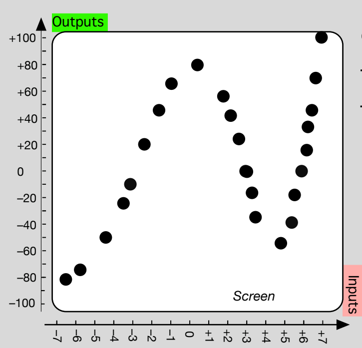
A function given by an I-O plot cannot of course return ∞.

3. Solving backward problems Solving backward problems, that is locating the input(s), if any, for which the function returns a given output, goes again exactly the same way as with solving **??**, again keeping in mind that Data-plots are sparse:.

PROCEDURE 2.2 To get x_0 for a given y_0 when f is given by an I O-plot

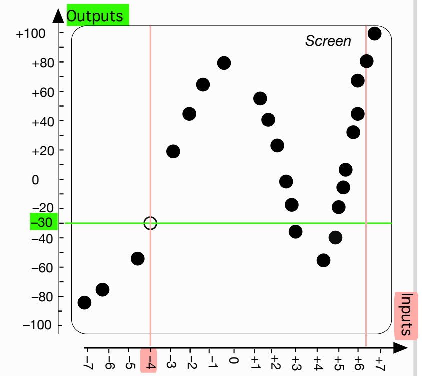
i. Tickmark y_0 on the output ruler,
ii. Draw an output level-line through y_0,
iii. Mark the relating dot(s), if any, where the output level-line intersects the I O-plot
iv. Draw an input level-line through each marked relating dot,
v. Read x_0 where the input level-line(s) intersect the input ruler.

DEMO 2.2a
$\mathcal{RON}$ being given by the I O-plot



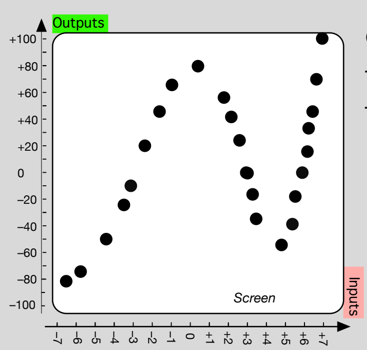
Get the input(s), if any, for which $\mathcal{RON}$ returns the output -30.

i. We mark the output-number -30 on the *output ruler*,
ii. We draw an output level-line through the mark,
iii. We mark the plot dot(s), if any, at the intersection of the output level-line with the I O-plot
iv. We draw an input level-*line* through the relating dot(s), if any,
v. The input-number(s), if any, is/are at the intersection(s), if any, of the input level-line(s), if any, with the input ruler: -4



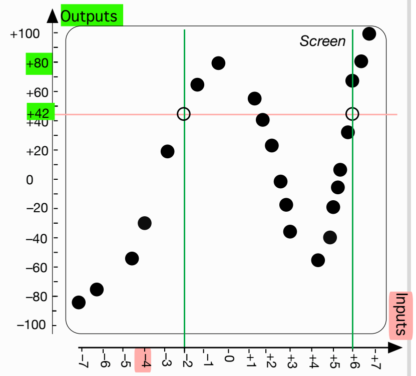
DEMO 2.2b

$\mathcal{MAE}$ being given by the I O-plot

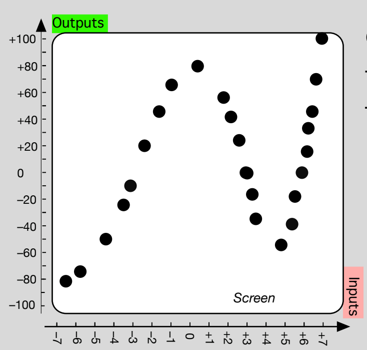


Get the input(s), if any, for which $\mathcal{MAE}$ returns the output -30.

i. We mark the output-number -30 on the *output ruler*,
ii. We draw an output level-line through the mark,
iii. We mark the plot dot(s), if any, at the intersection of the output level-line with the I O-plot
iv. We draw an input level-*line* through each relating dot(s), if any,
v. The input-number(s), if any, is/are at the intersection(s), if any, of the input level-line(s), if any, with the input ruler: -4, $+3$, $+5$



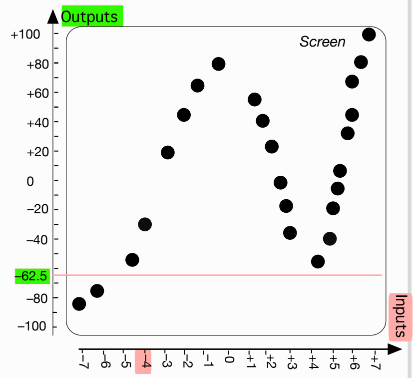
DEMO 2.2c $\mathcal{SALLY}$ being given by the I O-plot



Get the input(s), if any, for which $\mathcal{SALLY}$ returns the output -62.5.

pole

i. We mark the output-number -62.5 on the *output ruler*,
ii. We draw an output level-line through the mark,
iii. We mark the plot dot(s), if any, at the intersection of the output level-line with the I O-plot
iv. There is no intersection therefore there is no input level-*line* through the relating dot(s), if any,
v. The input-number(s), if any, is/are at the intersection(s),therefore there is no input-number.



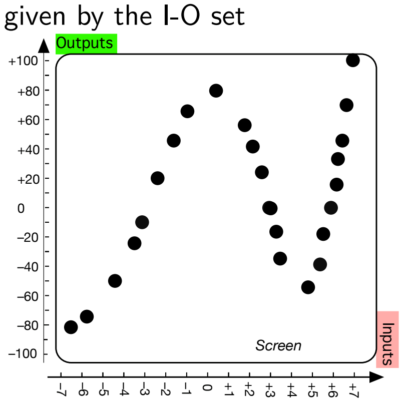
4. Zeros. The fact that backward problems usually have no solution because I O-plots are sparse is particularly unfortunate when we are looking for the zero(s) of a given function, that is the inputs for which the function returns 0 as output.

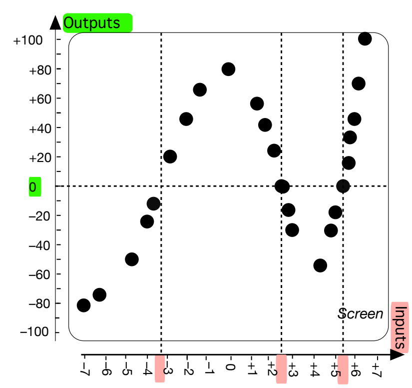
Zeros will be important because, as RBC will see, inputs whose output is 0 often separate inputs whose output is positive from inputs whose output is negative.

And, even though **??** (**??** **??**, **??**), a zero is a regular input.

However, with functions given by I-O plots, *RBC* will have to keep even more seriously in mind that **??** (**??** **??**, **??**).

Example 2.13. The function $\mathcal{EMMY}$ given by the I-O set





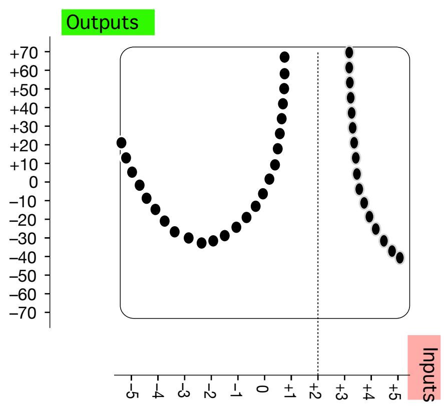
has two zeros

but it certainly looks like $\mathcal{EMMY}$ also has a zero between -3 and -4

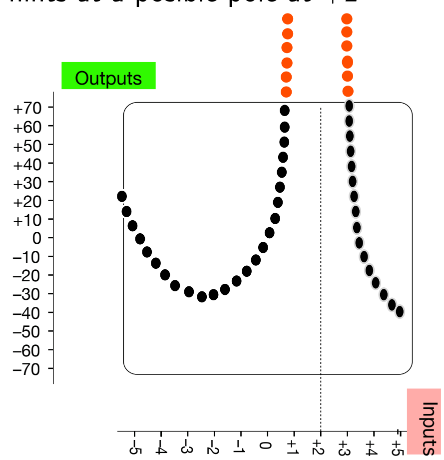
5. Poles. An even more important backward problem will be locating the **pole(s)** if any, of a function, that is the inputs for which the function returns ∞ as output. gradual

Of course, a pole is *not* a regular input since a function given by a I-O plot cannot have pole(s) since all the outputs are medium-size numbers. Yet, I-O plots can hint at possible pole(s).

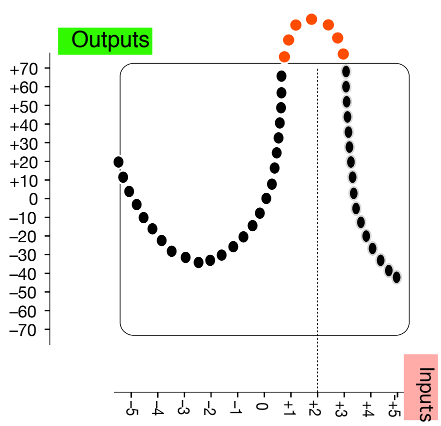
Example 2.14. It might seem that the I-O set

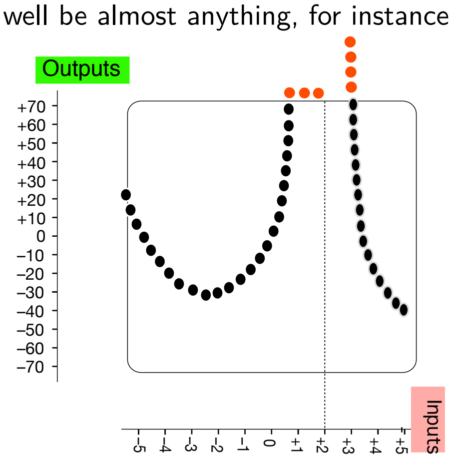


hints at a posible pole at $+2$



but of course the I-O set could equally well be almost anything, for instance





6. Discrete Calculus. Calculus is To See Change but there are several difficulties with discrete functions:

i. Since collections of left numbers are sparse, the changes with discrete functions are not **gradual** as the relating-dots are therefore also sparse. In fact, discrete functions cannot return any output for most inputs.

dot-interpolate
intermediate relating dot

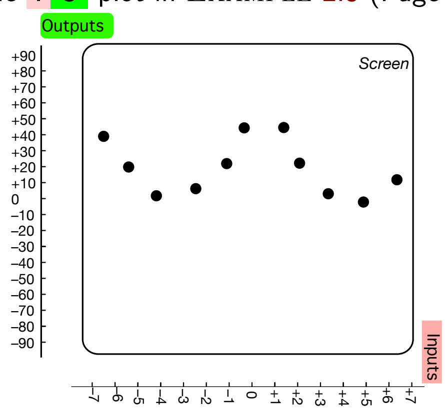
So, while the DISCRETE CALCULUS[24] which deals with discrete functions is a very important part of MATHEMATICS, *RBC* discussed functions given by I O -plots only for introductory purposes and will deal only with functions where the changes are mostly gradual.

For instance, one can reset the plotter and make another run.

ii. Nevertheless, it is worthwhile saying a few words about **dot-interpolation**[25], that is the creation of **intermediate relating dots**. The trouble with dot-interpolations, though, is that just about anything can happen with intermediate relating dots:

a. There is no guarantee that the dot-interpolated I O -plot will still meet the **??** (**?? ??**, **??**).

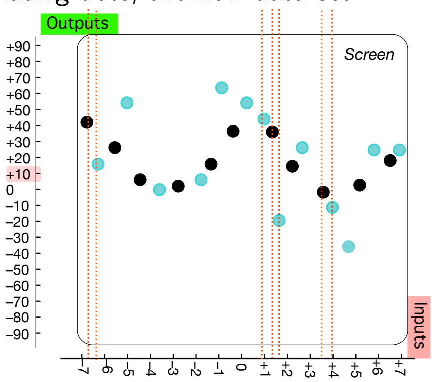
EXAMPLE 2.15. The I O -plot in EXAMPLE 2.3 (Page 94),



meets the **??** (**?? ??**, **??**) but:

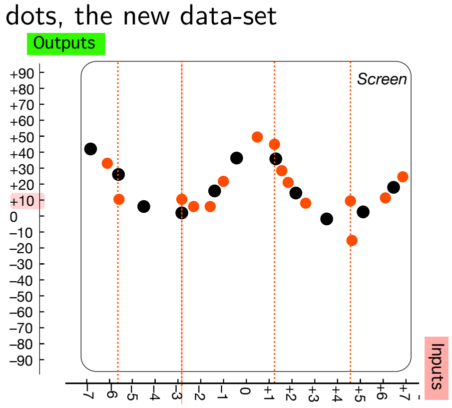
[24] https://en.wikipedia.org/wiki/Discrete_calculus
[25] https://en.wikipedia.org/wiki/Interpolation

while with the *blue* intermediate relating dots, the new data-set



still gives a function,

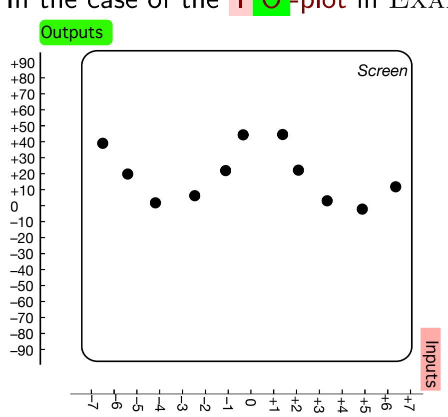
with the *red* intermediate relating dots, the new data-set



does *not* give a function any more.

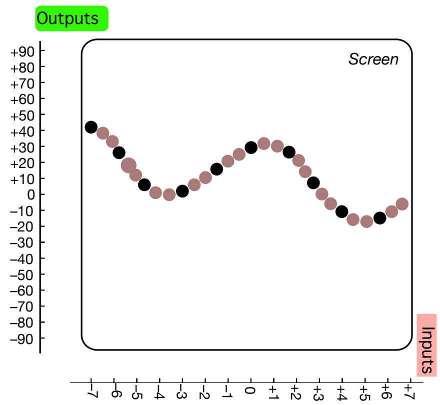
b. Even wnen the dot-interpolated I O-plot *does* give a function, that function can be just about *any* function

EXAMPLE 2.16. In the case of the I O-plot in EXAMPLE 2.3 (Page 94)

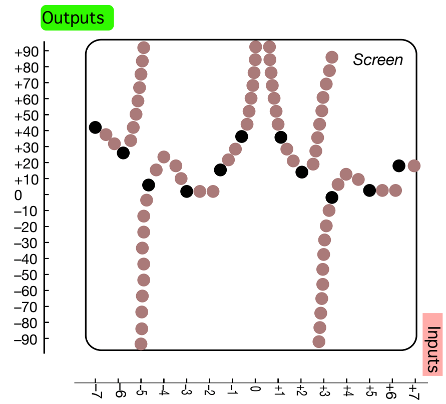


the following two dot-interpolations both give a function but

While the intermediate relating dots could of course be:



the intermediate relating dots could just as well be:



In fact, *how* to dot-interpolate an I O -plot is not at all a simple matter and there are many methods for coming up with *likely* outputs for missing intermediate inputs[26].

iii. Another difficulty with I O -plots is that functions given by I O -plot can involve only *finite* numbers whereas, in sciences and engineering, CALCULUS needs to deal also with:

- *infinitesimal* numbers in order to consider the neighborhoods of given finite numbers to take experimental imprecision into account,

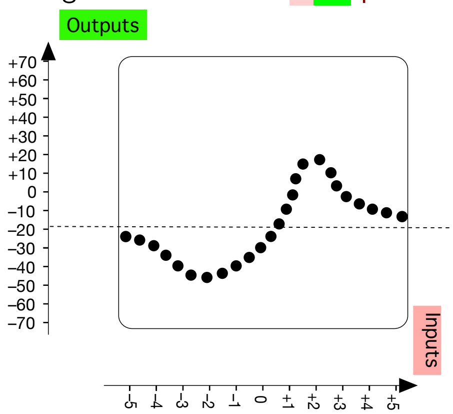
and

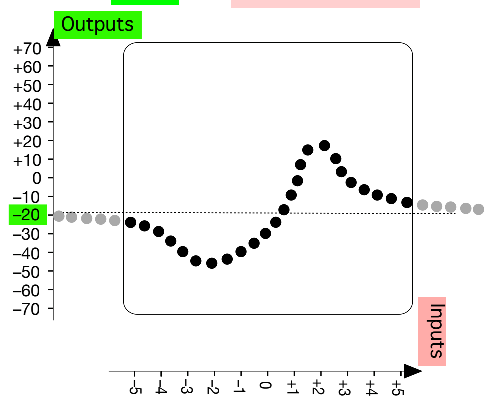
- *infinite* numbers in order to consider changes in the long haul.
  Since only pairs of finite numbers can be plotted, when the given input is inputs *near* infinity , an I O -plot cannot provide any information about the outputs for inputs *near* infinity , namely *large-size* input-numbers . However, occasionally, the I O -plot can *hint* at what the function *might* return for inputs *near* infinity

EXAMPLE 2.17.

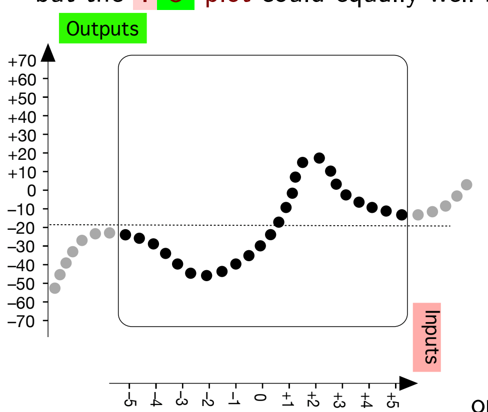
[26] https://en.wikipedia.org/wiki/Interpolation

It might seem that the I O-plot hints at -20 for inputs near ∞:

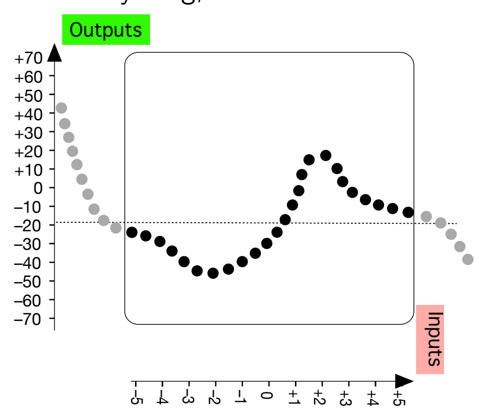




but the I O-plot could equally well be almost anything, for instance

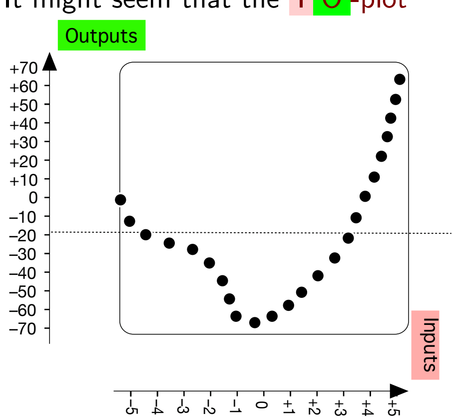


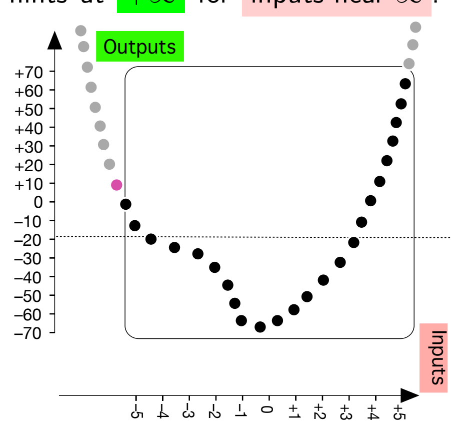
or



EXAMPLE 2.18.

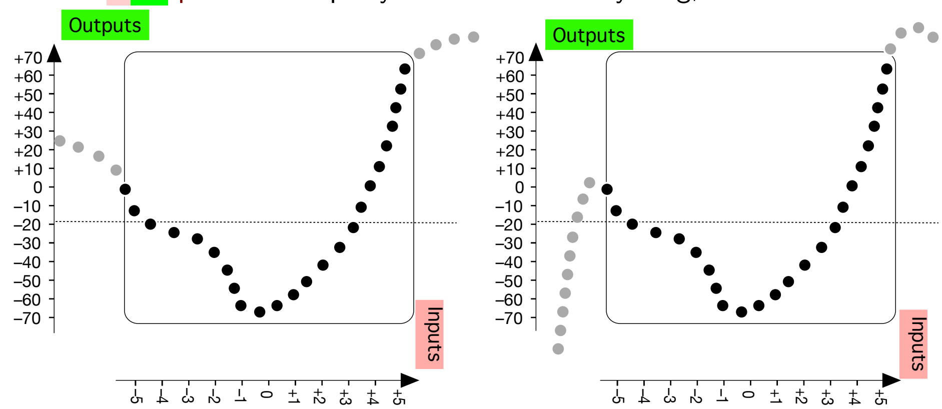
It might seem that the I O-plot hints at $+\infty$ for inputs near ∞:





curve
extended Cartesian setup
offscreen
finite input
infinite input
finite output
infinite output
Mercator view

but the I O-plot could equally well be almost anything, for instance



Thus, the DISCRETE CALCULUS cannot really deal with changes.

3 Functions Given By Curves

Don't worry, you don't have to know the calculus meaning of the word curve and you can go by just the ordinary English meaning.

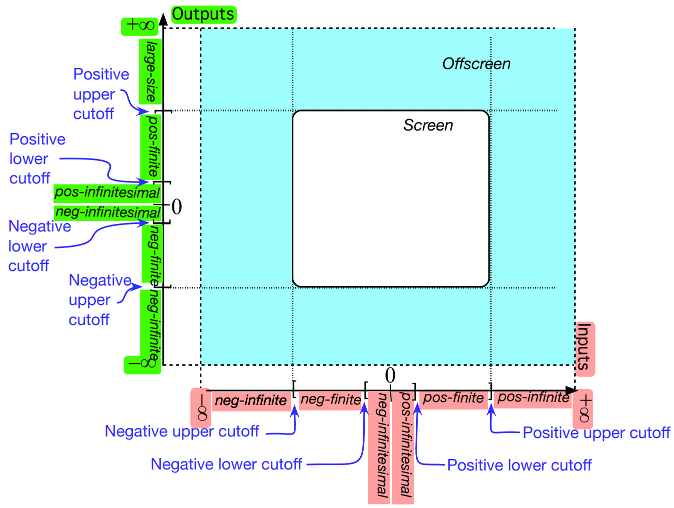
In order for *RBC* to deal with changes, functions will have to be given by a **curve**[27]. but then the Cartesian setup will have to be an **extended Cartesian setup**, that is a Cartesian setup that:

- Employs 2 pt compactifications of qualitative rulers in order to picture neighborhoods including a neighborhood of 0 and a neighborhood of 0
- Includes an **offscreen** space around the screen to provide a neighborhood of ∞ and a neighborhood of ∞ with:
  - The upper cutoffs for **finite inputs** lined up vertically with the *left* and *right* sides of the screen so that finite inputs will be below the screen and **infinite inputs** will be in a neighborhood of ∞ below the offscreen,
  - The upper cutoffs for **finite outputs** lined up horizontally with the *bottom* and *top* of the screen so that finite outputs will be left of the screen and **infinite ouiputs** will be in a neighborhoods of ∞ left of the offscreen.

As Descartes might have drawn it had he thought of infinite numbers.

1. Mercator view. By far the simplest way to picture an extended Cartesian setup is by way of a **Mercator view**[28] which is just a flat view

[27] https://en.wikipedia.org/wiki/Curve
[28] https://en.wikipedia.org/wiki/Mercator_projection

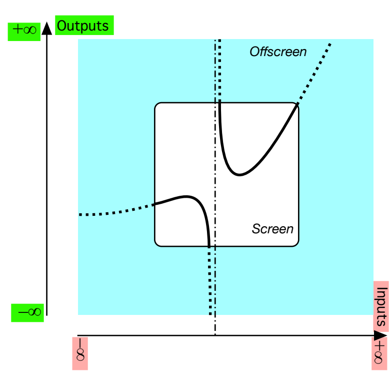


global graph
onscreen graph
offscreen graph

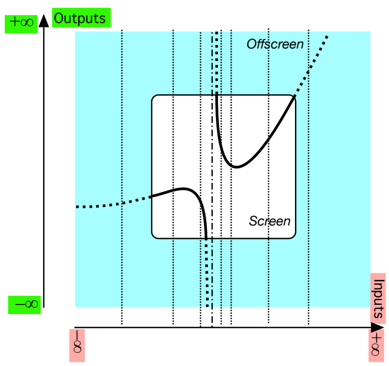
Then, whenever a given curve meets the Functional Requirement (DEFINITION (Restated) 2.1, Page 102), *RBC* will employ **global graph** for the whole curve and:

- **onscreen graph** for the part of the curve that is on the screen,
- **offscreen graph** for the part of the curve that is offscreen,

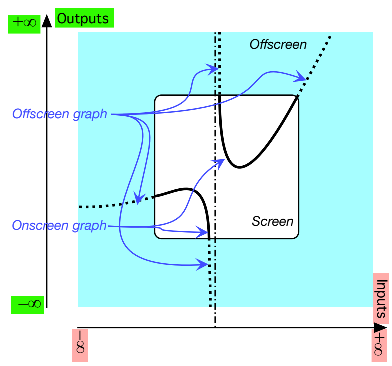
EXAMPLE 2.19. The curve



satisfies the Functional Requirement:



and so the curve is the global graph of a function:



The problem is a difficult one and Mercator's solution was the first in a long list[29].

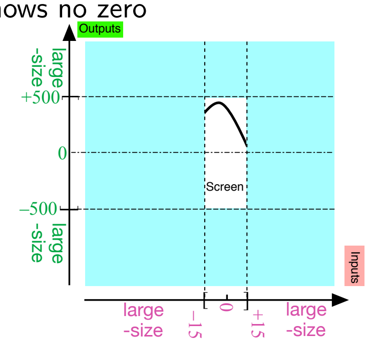
2. Limitations of the Mercator view. Even though the Mercator view is by far the most commonly employed, it is important to be aware of the severe limitations to the information which Mercator views can provide about a function.

i. How much an *onscreen* graph shows about a function depends *very much* on the cutoff sizes for finite numbers.

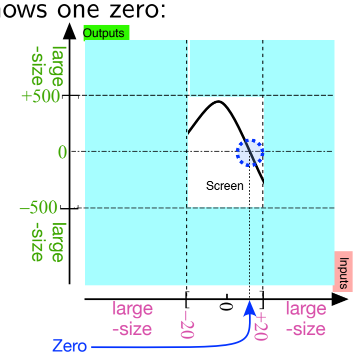
For instance, Mercator views do not necessarily show all the zeros of a function.

EXAMPLE 2.20. The following *onscreen* graphs of the function $\mathcal{ZANY}$ are all at the same scale and differ only by the cutoff size for finite input numbers:

With the cutoff for finite input numbers at 15, the onscreen graph shows no zero



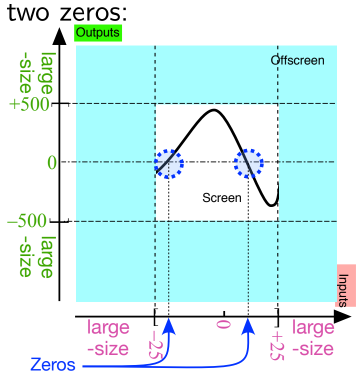
With the cutoff for finite input numbers at 20, the onscreen graph shows one zero:



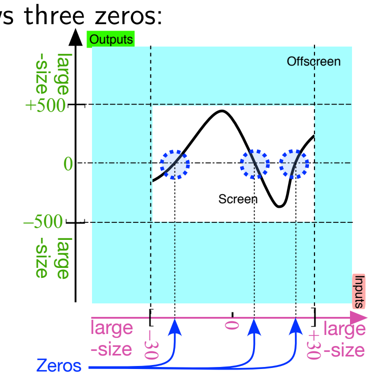
[29] https://en.wikipedia.org/wiki/List_of_map_projections

conclusive

With the cutoff for finite input numbers at 25, the onscreen graph shows two zeros:



With the cutoff for finite input numbers at 30, the onscreen graph shows three zeros:



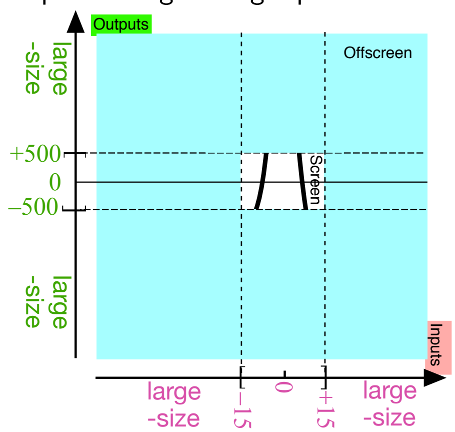
In other words, the Mercator views of a given function are *not* **conclusive** as to the zeros of that function.

ii. How much an *onscreen* graph shows about a function depends also *very much* on the cutoff size for finite outputs.

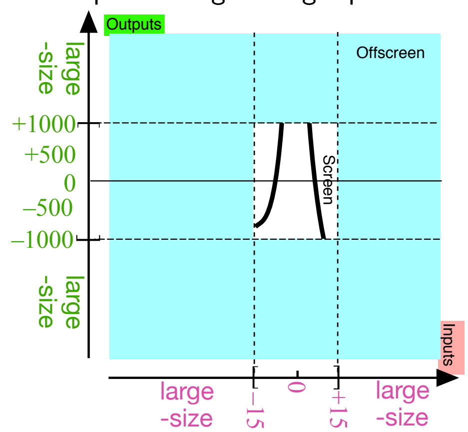
For instance, another very important backward problem will be locating the pole(s), if any, of a function, that is those inputs for wich the function returns ∞ but of course Mercator views cannot do that.

EXAMPLE 2.21. The following onscreen graphs of the function $\mathcal{COTY}$ are all at the same scale and differ only by the cutoff size for finite output numbers:

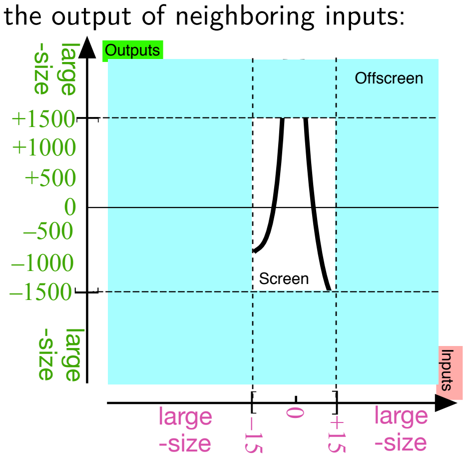
With the cutoff size for finite output numbers at 500, the onscreen graph does not show whether or not there is an input between -15 and $+15$ whose output is larger than the output of neighboring inputs:



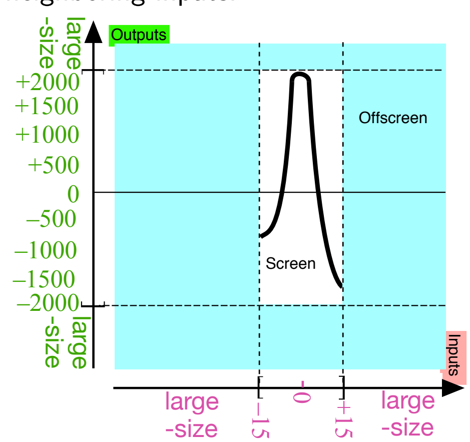
With the cutoff size for finite output numbers at 1000, the onscreen graph still does not show whether or not there is an input between -15 and $+15$ whose output is larger than the output of neighboring inputs:



With the cutoff size for finite output numbers at 1500, the onscreen graph still does not show whether or not there is an input between -15 and $+15$ whose output is larger than the output of neighboring inputs:



With the cutoff size for finite output numbers at 2000, the onscreen graph does show there is an input between -15 and $+15$ whose output is larger than the output of neighboring inputs:



In other words, the Mercator views of a given function are *not* necessarily conclusive as to the inputs whose output is larger than the output of nearby inputs.

tube view

Altogether then:

Caution 2.2 On-screen graphs are *not necessarily* **conclusive** as to the output(s), if any, for finite inputs.

Finally, since the purpose in this Part I - Functions Given By Data (Page 63) is *introductory*, *RBC* will employ curves to *give* functions but eventually, in Part II - Calculatable Functions (Page 197) and after, *RBC* will employ curves only to *picture* functions that will have been given otherwise. In any case,

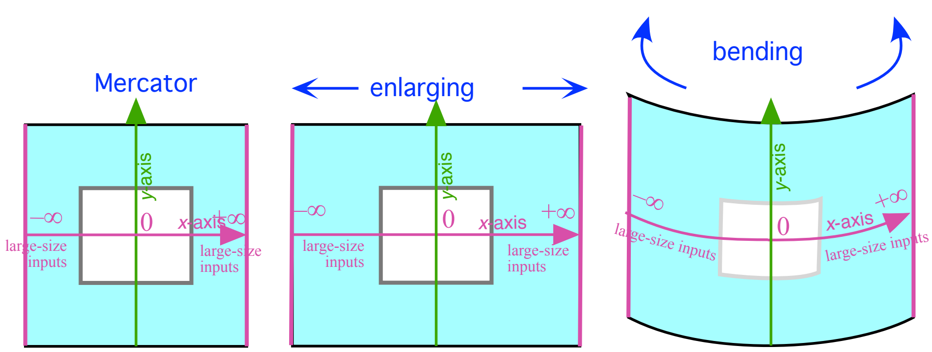
Caution 2.3 Functions given by curve are not necessarily simple and certainly not as simple as those employed here.

3. Compact views.

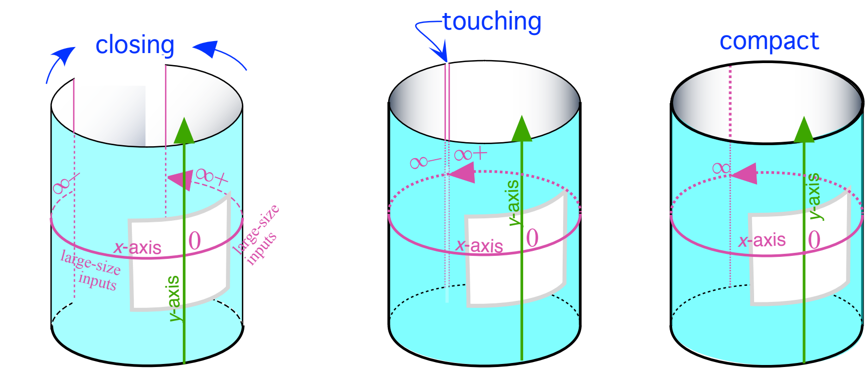
In order to show the off-screen graph which shows the 'behavior' of a function near poles, if any, and near infinity, *RBC* will employ several different compact views, that is views in which one both *axes* are compactified.

To see why axes *rather than rulers, just try to draw* rulers *in any of the compact views that follow.*

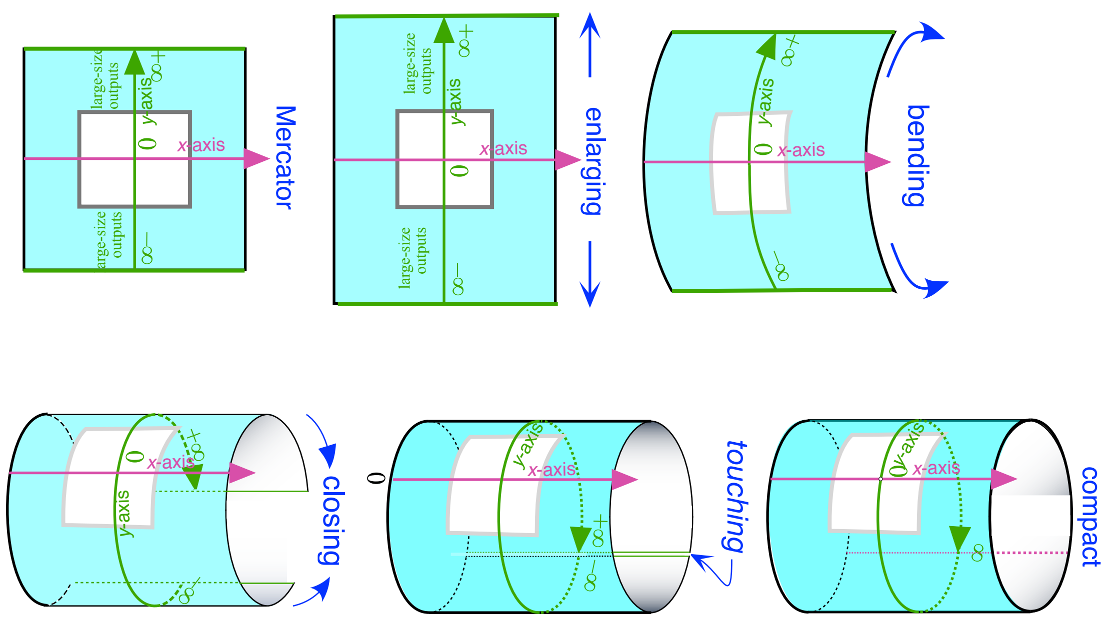
i. We can get a **tube view** by compactifying the *input* axis:



donut view

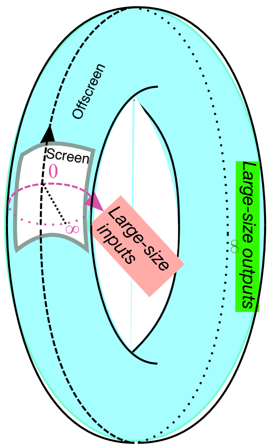


ii. We can get another kind of tube view by compactifying the *output* axis:

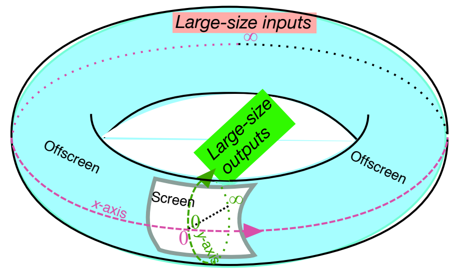


iii. We can get two kinds of **donut views** by compactifying the input axis and the output axis *one after the other*:

Magellan view

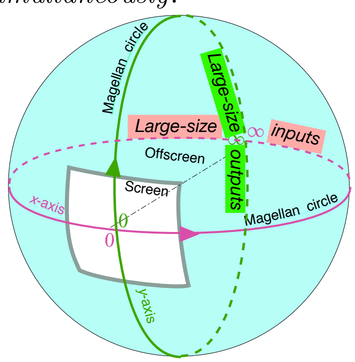


Input axis then output axis



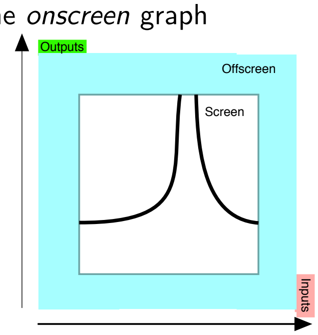
Output axis then input axis

iv. We can get a **Magellan view** by compactifying the input axis and the output axis *simultaneously*:



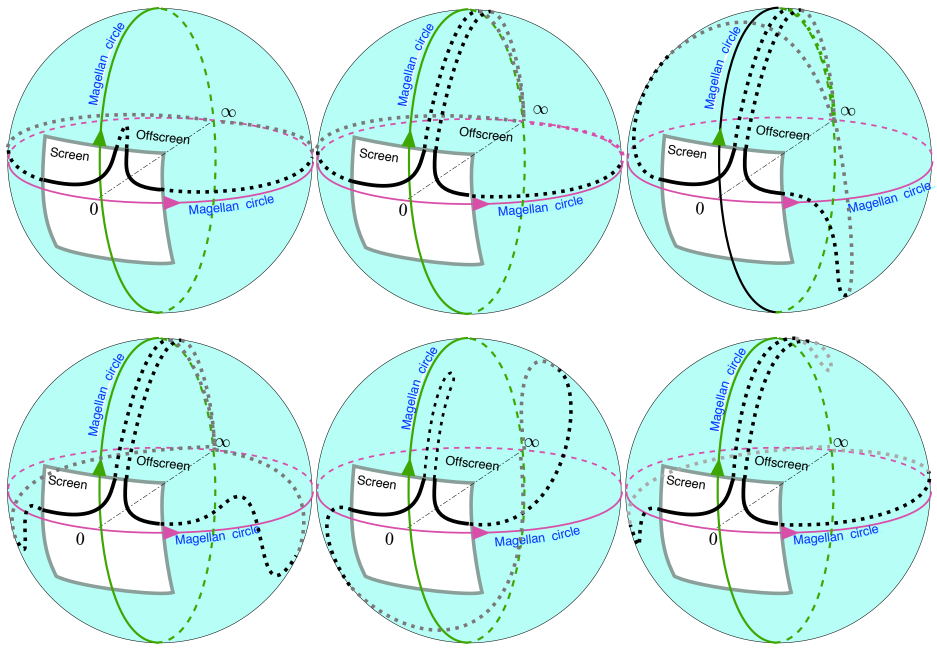
Magellan views are particularly good at showing why a Mercator view cannot *give* a function: different functions can have the same onscreen graph but different off-screen graphs.

EXAMPLE 2.22. The *onscreen* graph



smooth continuation

is the onscreen graph of any of the following functions viewed in Magellan view



as well as, in fact, many, many others.

OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR

4. OK so far - OK so far - OK so far - OK so far .

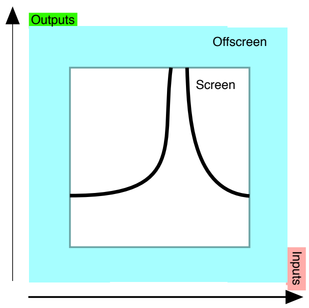
OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR

So, in order for a curve to be the onscreen graphof a function, *RBC* will make the following

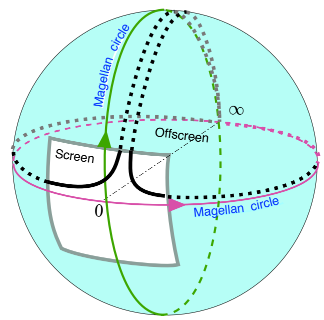
AGREEMENT 2.2 With functions given by curve, the upper cutoff size for finite input and finite outputs will be such that the off-screen graph is simply a **smooth continuation** of the onscreen graph. (However, with other types of functions, there are different kinds of continuations as, for instance, with the 'periodic' functions to be investigated in VOL. II.)

pole
parity
even pole
odd pole

EXAMPLE 2.23.
Given the onscreen graph in EXAMPLE 3.55 (Page 177):



by AGREEMENT 2.2 - (Page 122), the global graph can only be



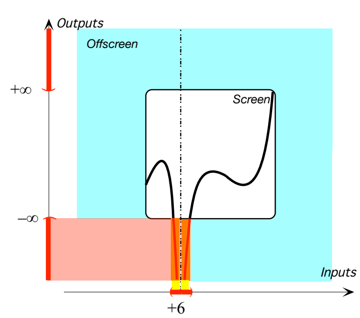
5. Pole of a function. See **??** (**??** **??**, **??**)

Given a function f, a **pole** of f is a *medium* input whose height-size is $\langle large, large\rangle$. We will distinguish two kinds of poles according to their parity:

We will distinguish two kinds of poles according to their **parity**:

- An **even pole** is a pole whose height-sign is either $\langle +, +\rangle$ or $\langle -, -\rangle$.

EXAMPLE 2.24. For the function f given by the global graph

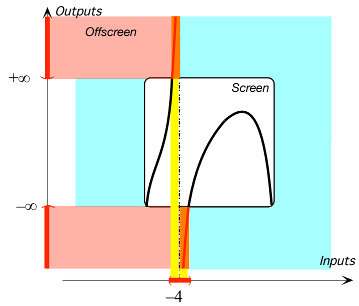


the medium input $+6$ is an *even pole* because:
- the *outputs* for inputs *near* $+6$ are all $large$,
- height-sign f near $+6 = \langle -, -\rangle$ (Same signs.)

- An **odd pole** is a pole whose height-sign is either $\langle +, -\rangle$ or $\langle -, +\rangle$.

curve-intepolate

EXAMPLE 2.25. For the function f given by the global graph



the medium input $+-4$ is an *odd pole* because:

- the *outputs* for inputs *near* -4 are all $large$,
- height-sign f near $-4 = \langle +, - \rangle$ (Opposite signs.)

6. Interpolating plots into curves?

7. Curve-Interpolating I-O plots. The next step beyond *dot*-interpolations of data-plots is ***curve*-intepolations** of data-plots[30].

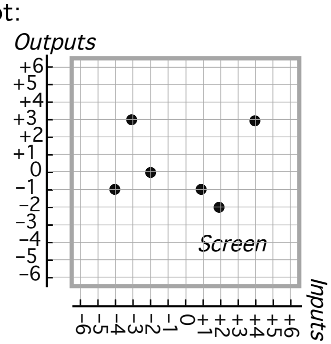
However, even though curve-interpolating data-plots tends to be much favored, curve-interpolation is even more risky than *dot*-interpolation.

THEOREM 2.1 In the absence of suplementary information, a function given by an I-O plot cannot be extended to a *single* function given by a curve.

Proof. Take an intermediate input and pair it with two different outputs. We can then curve-interpolate through either one of the two pairing dots. □

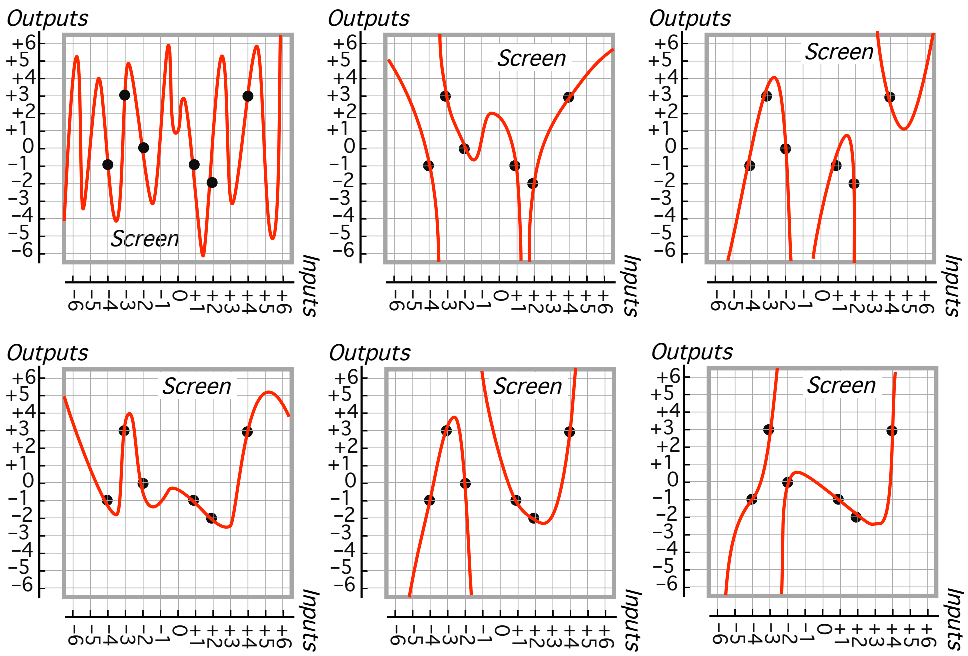
EXAMPLE 2.26. Suppose the function $\mathcal{RINO}$ was given by the following input-output table and therefore the following plot:

Inputs	-4	-3	-2	$+1$	$+2$	$+4$
Outputs	-1	$+3$	0	-1	-2	$+3$



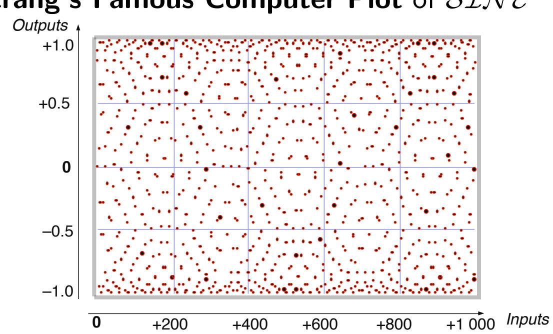
[30] https://en.wikipedia.org/wiki/Curve_fitting

Now, how should we join these plot dots? For instance:



And in fact, too many plot points can make it impossible to join them smoothly.

Example 2.27. The function $\mathcal{SINE}$ belongs to Volume II, but the point here is **Strang's Famous Computer Plot** of $\mathcal{SINE}$ [31]:



How are we to "join smoothly"?

Even mathematicians and scientists keep being amazed at the behavior of some of the functions that keep coming up.

[31]The plot appears on the back cover of Strang's *Calculus*, 1991, Wellesley-Cambridge Press, where it is discussed in Section 1.6 A Thousand Points of Light, pages 34-36.

input level-band
median line
width

8. Basic Expository Problem. So far, the reader would have every right to wonder how there could possibly be anything particularly complicated in dealing with functions but there are in fact many, many, functions that are unbelievably "complicated" (For a few examples, see https://www.google.com/search?q=Nowhere+continuous+function&client=firefox-b-d&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiY3Obx-9b9AhWeMlkFHZFDC-cQ_AUoAXoECAEQAw&biw=1012&bih=833&dpr=1.).

Basically, just about anything can happen.

The difficulty comes from the fact that **(i)** there is nothing in the **??** to prevent *any* outputs from being returned by a function for *any* inputs and so, in particular, nothing to prevent abrupt, huge, differences among outputs returned for even inputs that are near a given input, and that **(ii)** it is impossible to define anything like "complicated" functions—say for mathematicians—as opposed to "simple" functions for the rest of us.

The expository problem, then, is::

- If, to paraphrase Dudley, general statements about functions are rigorous so as to apply to *all* functions, including "complicated" functions—which the reader is not likely to encounter anytime soon, the reader is not likely, in Dudley's words, to "*see what is really going o*n".

On the other hand,

- Even if general statements about functions are worded so as to apply *only* to a few kinds of"simple" functions, then how is the reader to know when the functions will have become just too "complicated" for the general statement still to apply?

Our way out of this expository problem will be to agree that:

> **AGREEMENT 2.3** All general statements about functions will apply to *all* functions in *this* text. (As for functions in *other* texts, these statements may or may not apply.)

4 Local Graphs

We first need to introduce the equivalent of input level-*lines* and output level-*lines* for *neighborhoods* of given points.

1. Input level-band. An **input-level *band*** is "made up" of the input-level *lines* for the inputs in the neighborhood of the given input point.

In particular, the **median line** of the input level-*band* is the input level-*line* of the input point and the **width** of the input level-*band* is the width of the neigborhood of the given input point ..

The PROCEDURE, though, depends partly on whether the given input point is a given number x_0 or is ∞:

PROCEDURE 2.3 To get the input level-*band* for a neighborhood of an input point.

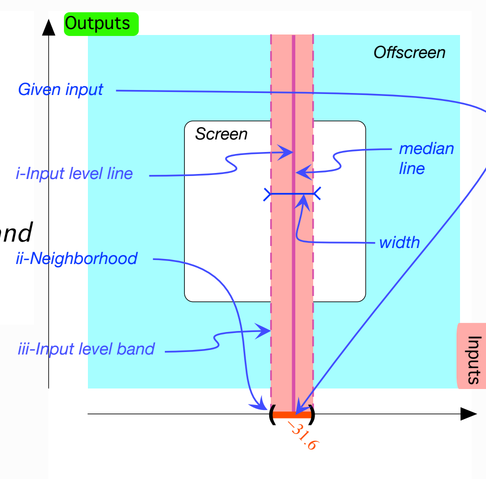
- When the input point is a given number x_0:
  i. Draw the input level-*line* for x_0,
  ii. Thicken the input level-*line* for x_0 to an input level-*band* for the *neighborhood* of x_0
- When the given point is ∞:
  i. Draw the input level-lines for $+\infty$ and for $-\infty$
  ii. Thicken the input level-lines for $+\infty$ and $-\infty$ into half level-bands corresponding to the width of the *half neighborhoods* of $+\infty$ and $-\infty$

DEMO 2.3a To get the input level-band for a neighborhood of the input number -31.6

i. We draw the input level *line* for -31.6

ii. We mark a *neighborhood* of -31.6 on the input ruler,

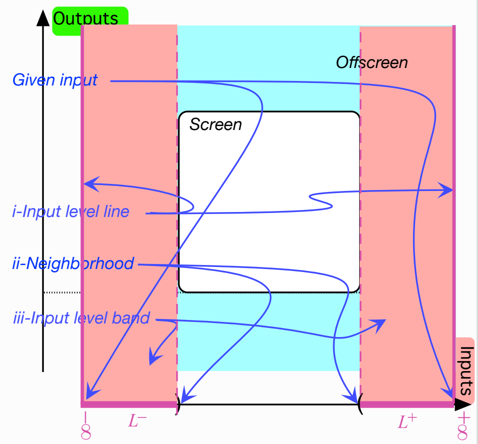
iii. We draw the input level-*band* with the width of the *neighborhood* of -31.6,



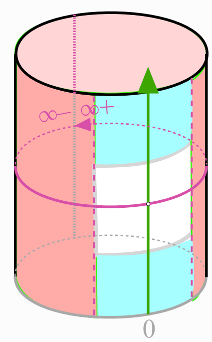
output level-band
median line
width

DEMO 2.3b To get the input level-band for a neighborhood of the input ∞

i. We draw the input level *lines* for $+\infty$ and $-\infty$
ii. We thicken ∞ into a neighborhood of ∞ (In Mercatot view),
iii. We thicken the input level-lines for $+\infty$ and $-\infty$ into rectangles corresponding to the width of the *half neighborhoods* of $+\infty$ and $-\infty$



In the above Mercator view, there appears to be *two* level-bands for ∞ but a tube view shows they are only the two sides of the input level-band near ∞:



2. Output level-band.

CAUTION 2.4

An **output level-band** is to a *neighborhood* of an output point, what the output level-line is to the output point itself.

In other words, the output-level *band* is "made up" of the output-level *lines* for the outputs in the neighborhood of the given output point. In particular, the **median line** of the output level-band is the output level-line of the output point and the **width** of the output level-band is the width of the neigborhood.

The PROCEDURE, though, depends partly on whether the given output point is a given number y_0 or ∞:

PROCEDURE 2.4 To get the output level-band for a neighborhood of an output point.

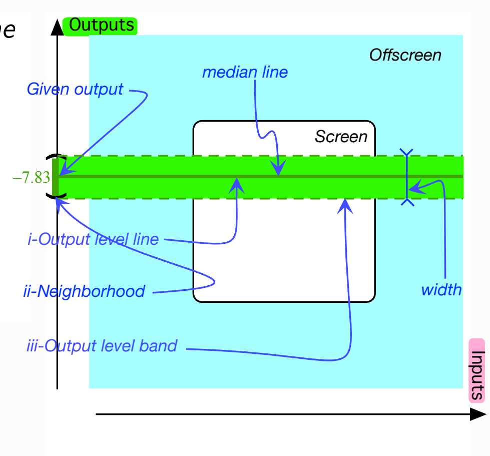
- When the output point is a number y_0:
  i. Draw the output level-line for y_0,
  ii. Thicken the output level-line for y_0 into an input level-band for the *neighborhood* of y_0
- When the given output point. is ∞:
  i. Draw the output level-lines for $+\infty$ and $-\infty$,
  ii. Thicken ∞ into a neighborhood of ∞ (In Mercaator view),
  iii. Thicken the output level-lines for $+\infty$ and $-\infty$ into rectangles corresponding to the width of the half *neighborhoods* of $+\infty$ and $-\infty$

DEMO 2.4a
To get the output level-band for a neighborhood of the output number -7.83

i. We draw the output level *line* for -7.83

ii. We mark a *neighborhood* of -7.83 on the output ruler,

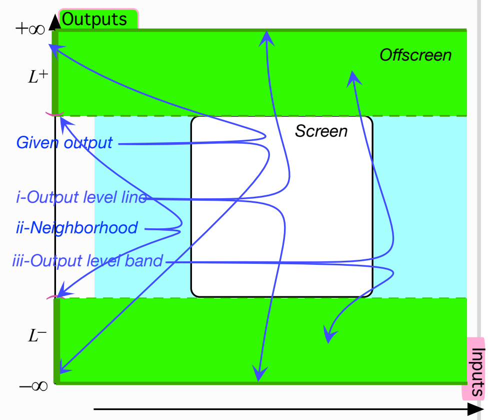
iii. We draw the output level-*band* with the width of the *neighborhood* of -7.83,



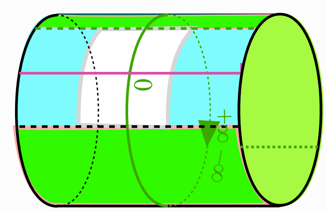
local frame

DEMO 2.4b
To get the output level-band for a neighborhood of the output point ∞

i. We draw the output level *lines* for $-\infty$ and $+\infty$
ii. We thicken ∞ into a *neighborhood* in Mercator view,
iii. We thicken the output level *lines* for $-\infty$ and $+\infty$ into rectangles the width of the half neighborhoods of $-\infty$ and $+\infty$



In the above Mercator view, there appears to be *two* level-bands for ∞ but a tube view shows they are only the two sides of the level-band near ∞:



3. Local frame. However, just like the plot dot for an *ordinary* input x_0, that is for an input-output pair of *numbers* $\left(x_0, y_0\right)$, is at the intersection of:

- the input level-*line* for the input *number* x_0
- the output level-*line* for the output *number* y_0,

similarly, the local graph for a neighborhood of a point will be within the **local frame** which is the intersection of:

- the input level-*band* for the neighborhood of the *input point*
- the output level-*band* for the neighborhood of the *output*point

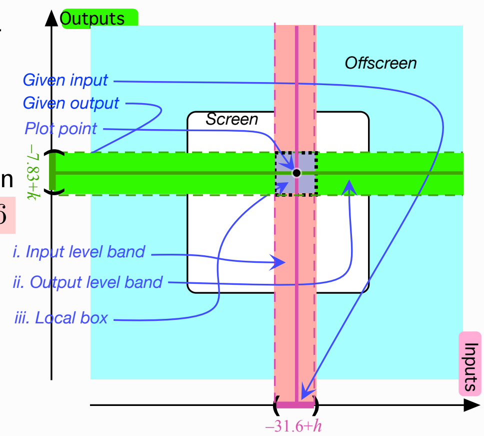
the input level-*band* and the output level-band:

PROCEDURE 2.5 To get the local frame for an $\left(\text{point}, \text{point}\right)$:
$\left(x_0, y_0\right)$ or $\left(x_0, \infty\right)$ or $\left(\infty, y_0\right)$ or (∞, ∞)

i. Get the input level-band for x_0 or ∞
ii. Get the output level-band for y_0 or ∞
iii. Frame the intersection of the input level-band and the output level-band

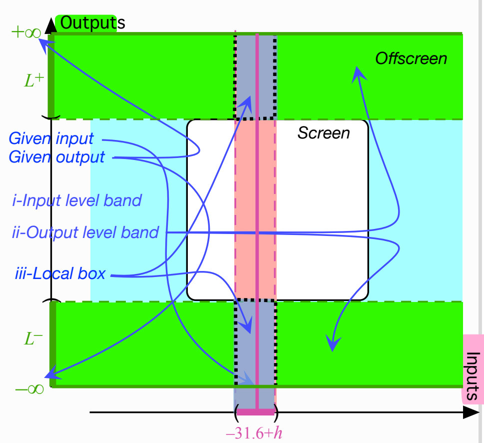
DEMO 2.5a
To get the local frame for the input-output pair $\left(-3.16, -7.83\right)$

i. We get the input level *band* for -3.16
ii. We get the output level *band* for -7.83
iii. We *local frame* the intersection of the input level-bands for -3.16 and -7.83



DEMO 2.5b
To get the local frame for the input-output pair $\left(-3.16, \infty\right)$

i. We get the input level *band* for -31.6
ii. We get the output level *band* for ∞
iii. We get the local frame for the intersection of the input level-band for -31.6 and the output level-band for ∞

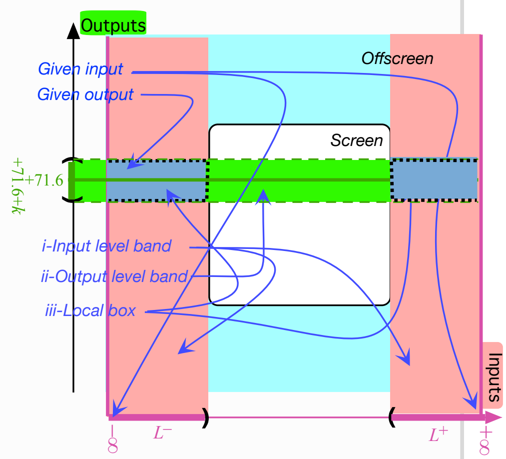


But then Magellan views are a lot harder to draw.

In the above Mercator view, there appears to be two local frames for ∞ but a donut view shows they are only the two halves of the same local frame.

Demo 2.5c
To get the local frame for the input-output pair of *numbers* $\left(\infty, +71.6\right)$

i. We get the input level *band* for ∞
ii. We get the output level *band* for $+71.6$
iii. We get the local frame for the intersection of the input level-band for ∞ and the output level-band for $+71.6$

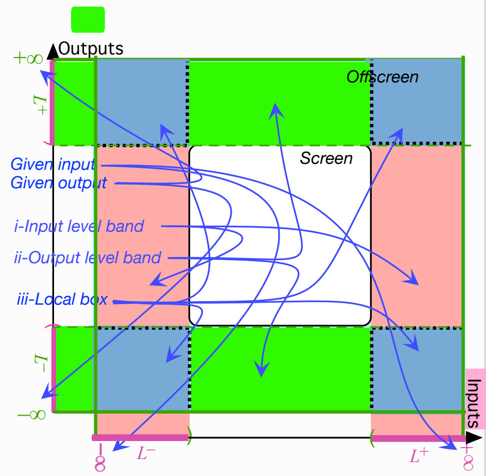


In the above Mercator view, there appears to be two local frames for ∞ but a donut view shows they are only the two halves of the same local frame.

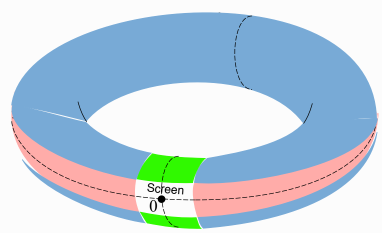
Demo 2.5d
To get the local frame for (∞, ∞)

local graph

i. We get the input level *band* for ∞
ii. We get the output level *band* for ∞
iii. We get the local frame for the intersection of the half input level-bands for ∞ and the half input level-bands for ∞



What appears to be four local frames are actually parts of *the* local frame because we are using a Mercator view instead of a Magellan view in which they would appear as the four quarters of a single local frame.



4. Local graph near a point Just the way a plot dot shows the inpui-output pair for a given input *number*, a **local graph** will show the inpui-output pairs for the input *numbers* in a *neighborhood* of a given input *point*:

Procedure 2.6 To get the local graph for inputs in a neighborhood of a given point when the function is given by a global graph

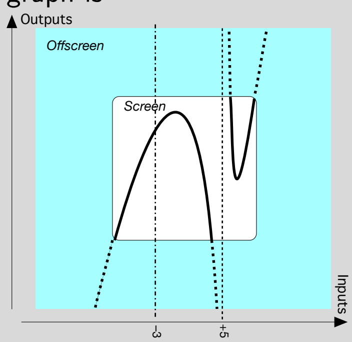
i. Mark a neighborhood of the point on the input ruler,
ii. Draw the input level-band for the neighborhood of the point using **?? ??** - **??** (**??**),
iii. The local graph near the point is at the intersection of the input level-band and the global graph.

While the procedure is the same regardless of the nature of the point, we will look at the difference cases separately

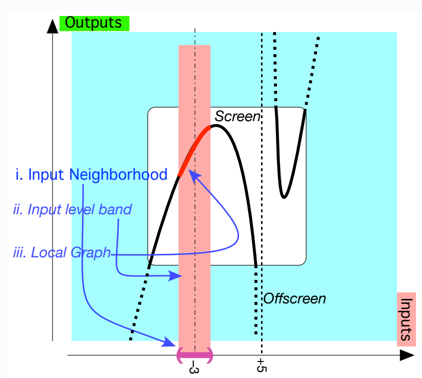
5. Local graph near x_0.

DEMO 2.6a

To get the local graph near -3 of the function $\mathcal{MARE}$ whose global graph is

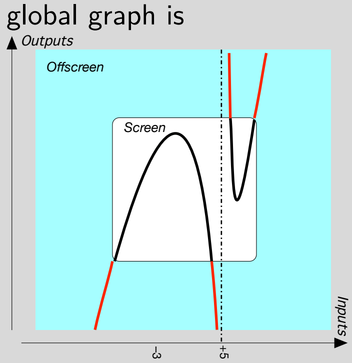


i. We mark a *neighborhood of* -3 on the *input ruler*,
ii. We draw the *input level* band for the *neighborhood* of -3,
iii. The *local graph* of $\mathcal{MARE}$ near -3 is at the intersection of the input level *band* with the global graph,

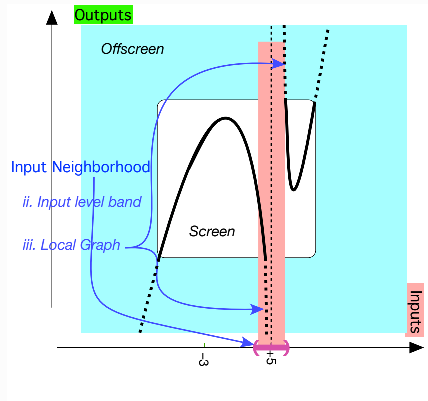


DEMO 2.6b

To get the local graph near the pole $+5$ of the function $\mathcal{JEN}$ whose global graph is



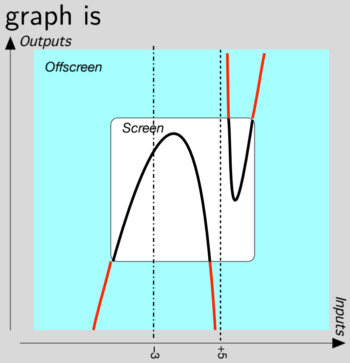
i. We mark a *neighborhood of* $+5$ on the *input ruler*,
ii. We draw the *input level* band through the *neighborhood* of $+5$,
iii. The *local graph* of $\mathcal{JEN}$ near $+5$ is the intersection of the input level *band* with the global graph,



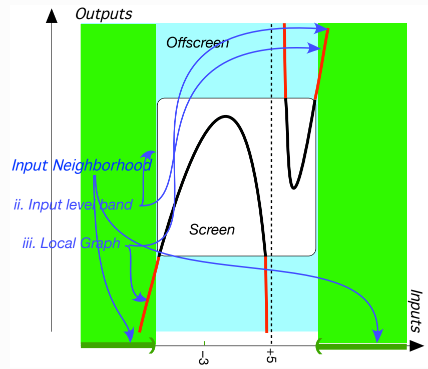
6. Local graph near ∞.

Keep in mind that even for large inputs, a function may return outputs of any qualitative size, medium-size: infinite or infinitesimal.

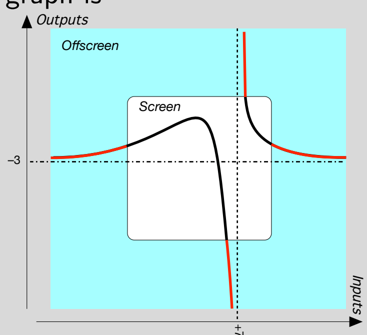
DEMO 2.6c
To get the local graph near ∞ of the function $\mathcal{REN}$ whose global graph is



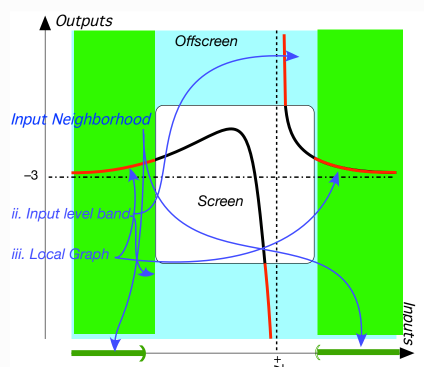
i. We mark a *neighborhood of* ∞ on the *input ruler*,
ii. We draw the *input level* band through the *neighborhood* of ∞,
iii. The *local graph* of $\mathcal{REN}$ near ∞ is the intersection of the input level *band* with the global graph,



Demo 2.6d
To get the local graph near ∞ of the function $\mathcal{MINA}$ whose global graph is

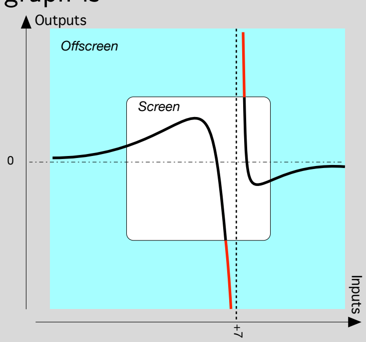


i. We mark a *neighborhood of* ∞ on the *input ruler*,
ii. We draw the *input level* band through the *neighborhood* of ∞,
iii. The *local graph* of $\mathcal{MINA}$ near ∞ is the intersection of the input level *band* with the global graph,

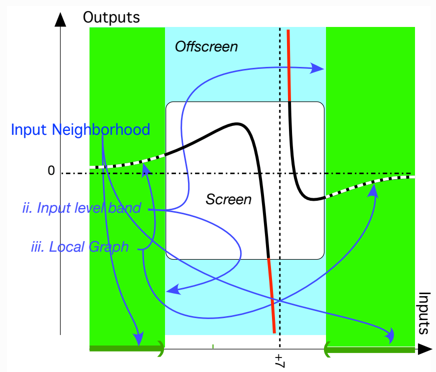


Demo 2.6e
To get the local graph near ∞ of the function $\mathcal{RHEA}$ whose global

graph is



i. We mark a *neighborhood of* ∞ on the *input ruler*,
ii. We draw the *input level* band through the *neighborhood* of ∞,
iii. The *local graph* of $\mathcal{RHEA}$ near ∞ is the intersection of the input level *band* with the global graph,



BeginWORKzone - BeginWORKzone - BeginWORKzone - BeginWORKzone - BeginWORKzone

7. Facing the neighborhood.

There is no reason to expect the local behavior of a function to be the same on both sides of a input point, be it x_0 or ∞, see Subsection 2.3 - Solving backward problems (Page 105)) and Subsection 2.5 - Poles (Page 108)).

In order to deal *separately* with each side of a neighborhood of a given point, we first need to state precisely which side of the given point is going to be LEFT of the given point and which side of the given point is going to be RIGHT of the given point.

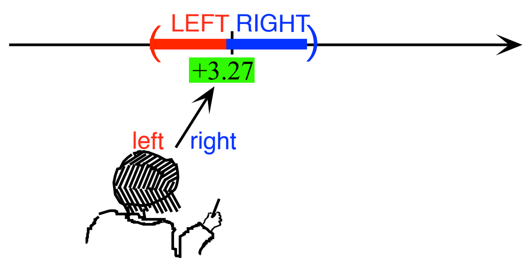
EXAMPLE 2.28. Given a neighborhood of the *number* $+3.27$, JILL can face the center of the neighborhood and then:

- what is to JILL's left will be what is LEFT of $+3.27$

and

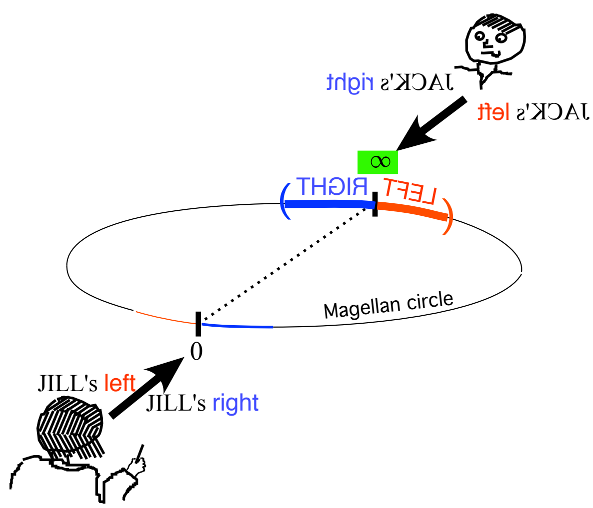
basic format angle

- what is to JILL's right will be what is RIGHT of $+3.27$.



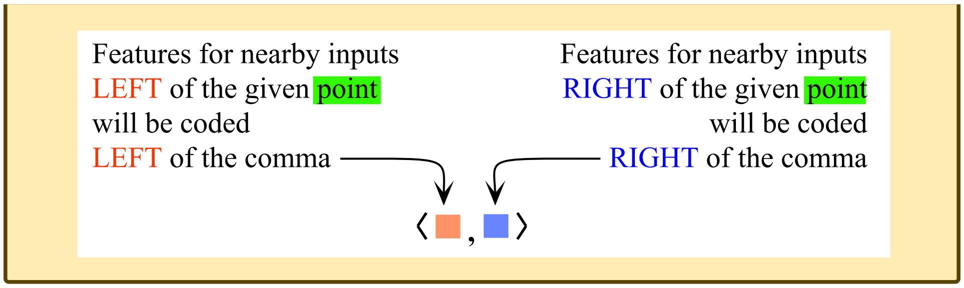
EXAMPLE 2.29. Given a neighborhood of ∞, JILL cannot face the center of the neighborhood and so, using a Magellan circle, she must imagine JACK facing a neighborhood of ∞ and then:
- what is to JACK's left will be what is LEFT of ∞

and
- what is to JACK's right: will be what is RIGHT of ∞

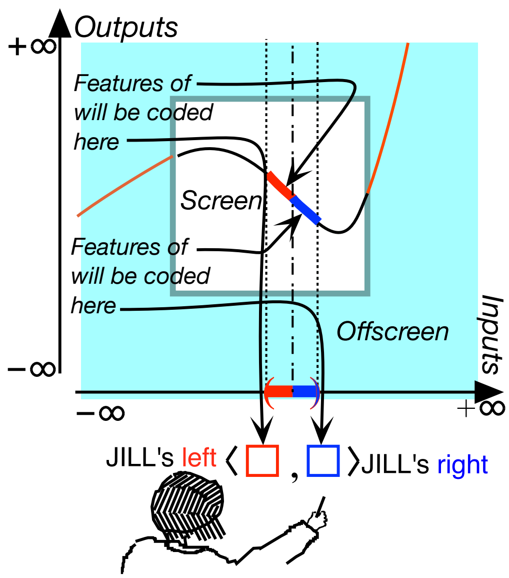


8. Local code. in order to describe *separately* the 'local behavior' on each side of the given input, we will use the following format:

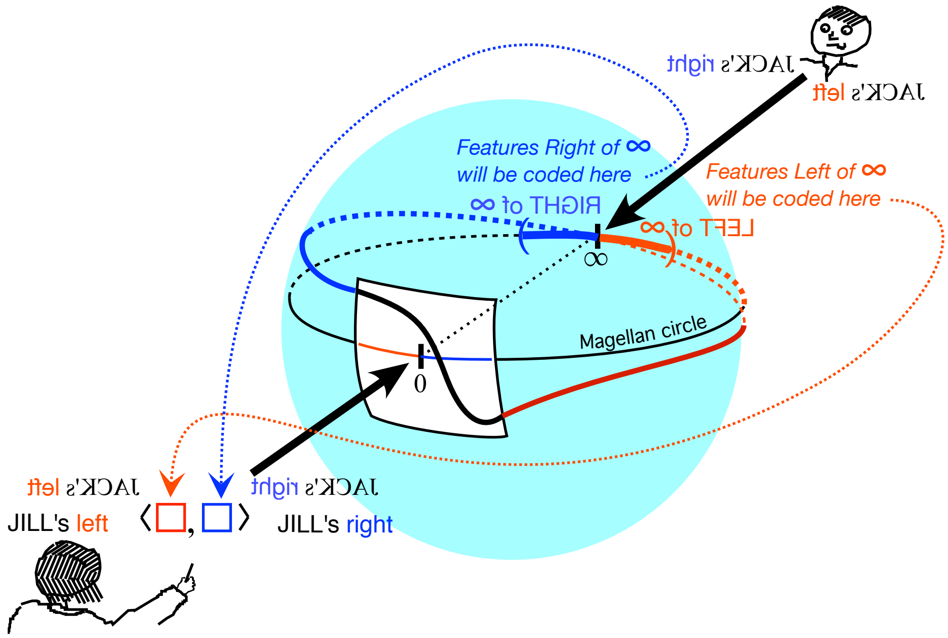
> **DEFINITION 2.3** To code the features of the local graph near a given point, we will write the codes for the feature on each side between two **angles** with a *comma* to separate the behaviors on the sides of the neighborhood of the given point:



EXAMPLE 2.30. When the local graph is near a number, JILL can face the center of the neighborhood:



EXAMPLE 2.31. When the local graph is near ∞ and since JILL can only imagine JACK facing *infinity* on the far side of a Magellan circel:


JACK's right
JACK's left
Features Right of ∞
will be coded here
Features Left of ∞
will be coded here
RIGHT of ∞
LEFT of ∞
∞
Magellan circle
0
JACK's left
JACK's right
JILL's left
JILL's right

height

Chapter 3

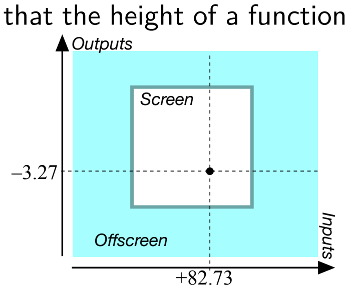
The Looks Of Functions

Finally, even though functions are usually *not* given by way of curves but by way of Input-Output Rules (Chapter 4, Page 197), in this chapter and the next one we will continue to give functions by way of curves because this will allow us to *see* all the outputs returned by the function for all the inputs in a neighborhood of a given input.

1 Height

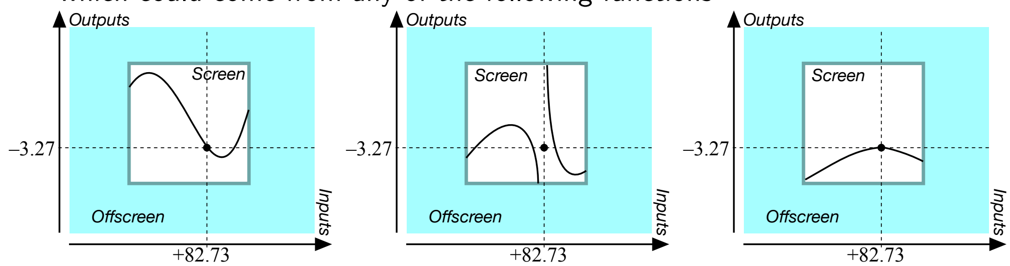
The **height** of a function f *at* a given *number* x_0 is just the *output* $f\left(x_0\right)$ provides almost no information about the graph of the function.

EXAMPLE 3.1. To say that the height of a function at $+82.73$ is -3.27 gives



local height-sign

which could come from any of the following functions



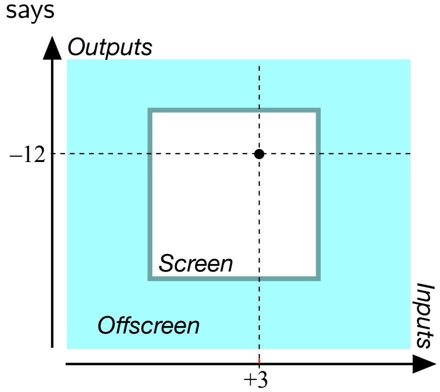
. . . and from many more.

1. Local height near a given point. Given a function f and given a point, the height of f *near* x_0 is

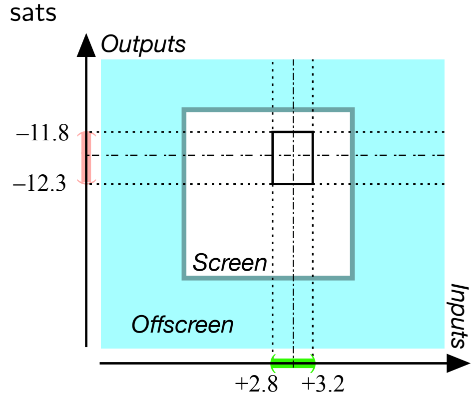
we want a thick version of the height of f *at* x_0 that is the height of f *near* x_0.

EXAMPLE 3.2. Given a function f, to say that

Height f *at* $+3 = -12$ says



Local height f *near* $+3 = -12 \oplus h$ sats



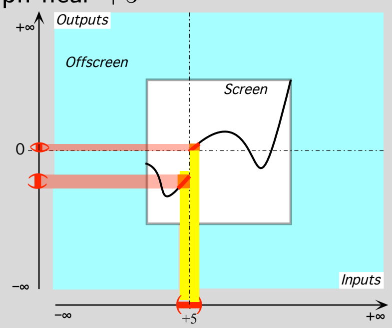
As will become clear why, though, we have to introduce and discuss the sign and the size of the local height separately.

2. Local height-sign. The **local height-sign** of f near x_0 is the sign, $+$ or $-$, of the outputs for nearby inputs on each side of the given input.

PROCEDURE 3.1 To get the local height-sign near x_0 of a function given by a curve, local height-size

i. Highlight the *local graph* near x_0 using **?? ??** - **??** (**??**)
ii. Get from the local graph the sign, $+$ or $-$, of the *outputs* for nearby inputs on each side of the given input,
iii. Code the local height-sign f using DEFINITION 2.1 - Functional Requirement (Page 94)

DEMO 3.1 To get the local height-sign near $+5$ for the function IAN from the local graph near $+5$



i. We get from the local graph the sign of the *outputs* for nearby inputs on each side of $+5$:

- The sign of the outputs *left* of $+5$ is $-$
- The sign of the outputs *right* of $+5$ is $+$

ii. We code the local height-sign:

Local height-sign IAN near $+5 = \langle -, + \rangle$

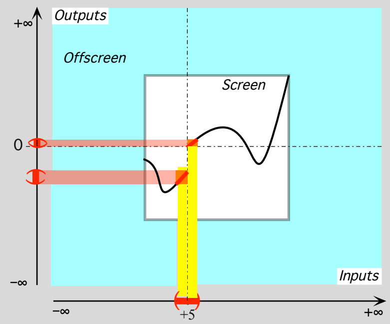
3. Height-size The **local height-size** of f near a given input is the qualitative size, *large*, *medium* or *small*, of the *outputs* for nearby inputs on each side of the given input.

PROCEDURE 3.2 To get the height-size near a given input of a function from its global graph,

i. Highlight the *local graph* near the given input using **?? ??** - **??** (**??**)

ii. Mark a neighborhood of the given point
iii. Get from the local graph the qualitative size, *large*, *medium* or *small*, of the *outputs* for nearby inputs on each side of the given input,
iv. Code height-size f according to **?? ??** - **??** (**??**)

DEMO 3.2a Get height-size near $+5$ for the function IAN from the local graph near $+5$



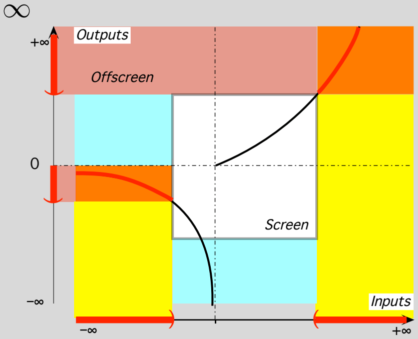
i. We get from the local graph the qualitative size, *large*, *medium* or *small*, of the *outputs* for nearby inputs on each side of $+5$:

- The size of the outputs *left* of $+5$ is *medium*
- The size of the outputs *right* of $+5$ is *medium*

ii. We code the height-size:

height-size IAN near $+5 = \langle medium, medium \rangle$

DEMO 3.2b Get height-size near ∞ for the function IAN from the local graph near ∞



parity
even zero
odd zero

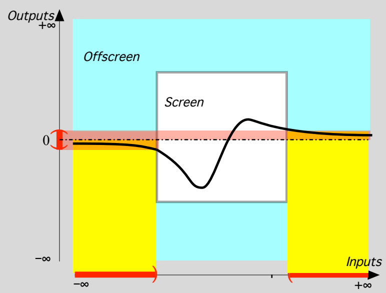
i. We get from the local graph the qualitative size, $large$, $medium$ or $small$, of the *outputs* for nearby inputs on each side of ∞:

- The size of the height *left* of ∞ is $large$
- The size of the height *right* of ∞ is $small$

ii. We code the height-size:

height-size IAN near $\infty = \langle large, small\rangle$

Demo 3.2c For the function



the Magellan input ∞ is a *zero* because:
the outputs for nearby inputs, both inputs *right* of ∞ and inputs *left* of ∞, are all $small$,

4. Parity of zeros and poles The height-size of a zero of a given function f is $\langle infinitesimal, infinitesimal\rangle$.

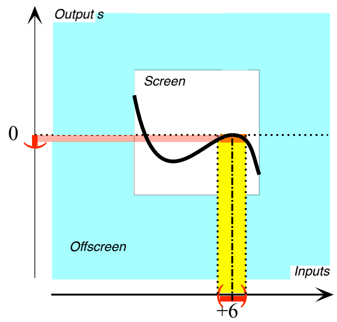
We will distinguish two kinds of zeros according to their **parity**:

An **even zero** is a zero whose height-sign is either $\langle +,+\rangle$ or $\langle -,-\rangle$.

An **odd zero** is a zero whose height-sign is either $\langle +,-\rangle$ or $\langle -,+\rangle$.

$x_{\infty\text{-height}}$
$x_{0\text{-height}}$
height

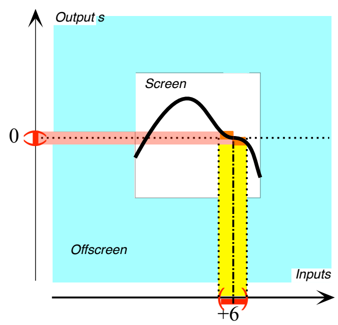
EXAMPLE 3.3. For the function f given by the global graph



the medium input $+6$ is an *even zero* because:

- the *outputs* for inputs *near* $+6$ are all $small$,
- height-sign f near $+6 = \langle -, - \rangle$ (Same signs.)

EXAMPLE 3.4. For the function f given by the global graph



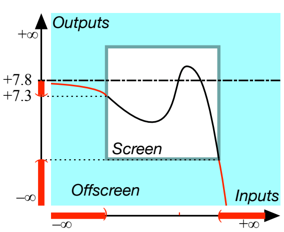
the medium input $+6$ is an *odd zero* because:

- the *outputs* for inputs *near* $+6$ are all $small$,
- height-sign f near $+6 = \langle +, - \rangle$ (Opposite signs.)

5. Local height near ∞ The concept of height provides us with conveniently systematic names:

- For a pole: $\boldsymbol{x}_{\infty\textbf{-height}}$
- For a zero: $\boldsymbol{x}_{0\textbf{-height}}$

The height near ∞



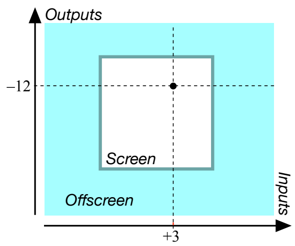
is $-large$ for inputs left of ∞ and $-small$ for inputs right of ∞

Given a function f, we will thicken the output AT a given input, be it x_0 or ∞, into the **height** *near* the given input.

Height height continuous at x_0

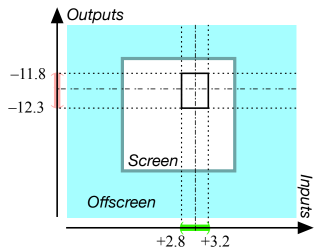
EXAMPLE 3.5.

The output at $+3$



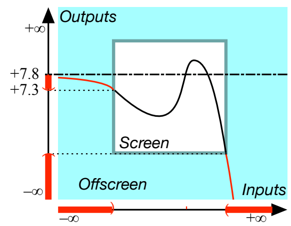
is -12

The height near $+3$



is $-12 \pm small$

The height near ∞



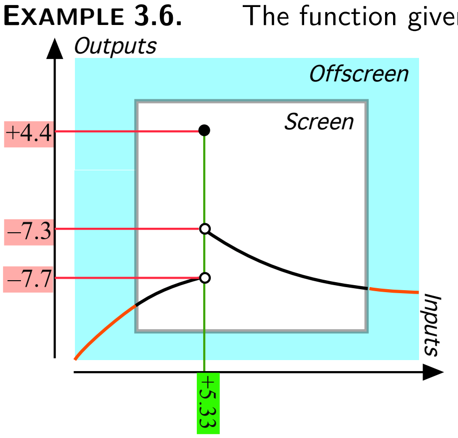
is $-large$ for inputs left of ∞ and $-small$ for inputs right of ∞

2 Height-continuity

The first kind of abrupt change that can occur is in the size of the outputs for nearby inputs.

For instance, we might expect that the outputs for inputs near a given input will have outputs that are near the output for the given input but, while this is often the case, this is absolutely not *necessarily* the case.

EXAMPLE 3.6. The function given by the global graph



- Sends $+5.33$ to a *positive* number, $+4.4$

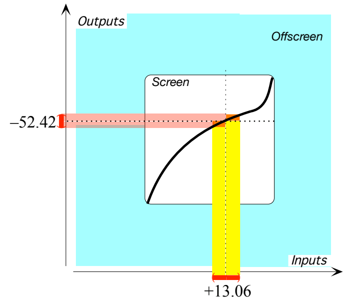
but

- Sends *all* other numbers to *negative* numbers .

1. Height-continuity at x_0. Given a medium-size input x_0, we tend to expect that functions will be **Height height continuous at x_0**, that is that the outputs for nearby inputs will themselves be near $f(x_0)$, the output at x_0.

height discontinuous
height discontinuous at x_0
jump
hollow dot

EXAMPLE 3.7.
The function



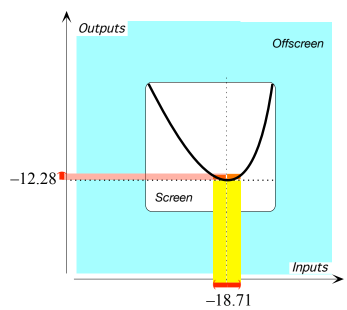
is *height continuous* at $+13.06$ because:

- the output *at* $+13.06$ is -52.42

and

- the outputs for *all* nearby Inputs, both *left* of $+13.06$ and *right* of $+13.06$, are themselves near -52.42.

EXAMPLE 3.8.
The function



is *height continuous* at -18.71 because

- the output *at* -18.71 is -12.28

and

- the outputs for *all* nearby Inputs, both *left* of -18.71 and *right* of -18.71, are themselves *near* -12.28.

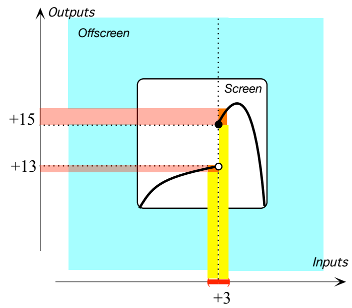
2. Height-discontinuity at x_0. Given a medium-size input x_0, a function is **height discontinuous at x_0** when *not all* the outputs for nearby inputs are near $f(x_0)$, the output at x_0.

- A function can be height discontinuous at x_0 because the function has a **jump** at x_0, that is because the outputs for nearby inputs on one side of x_0 are all near one medium-size output while all the outputs for nearby inputs on the other side of x_0 are near a different medium-size output.

  Since we use solid dots to represent input-output pairs, we will use **hollow dots** for points that *do not* represent input-output pairs.

EXAMPLE 3.9.

The function



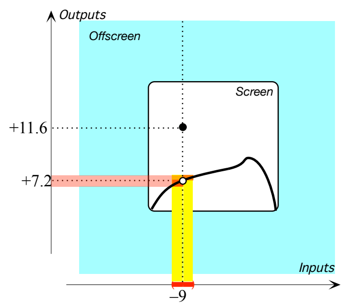
is *height discontinuous* at $+3$ because the function has a *jump* at $+3$ that is: gap

- the outputs for nearby inputs *right* of $+3$ are all near $+15$,

but

- the outputs for nearby Inputs *left* of $+3$ are all near $+13$.

EXAMPLE 3.10.
The function

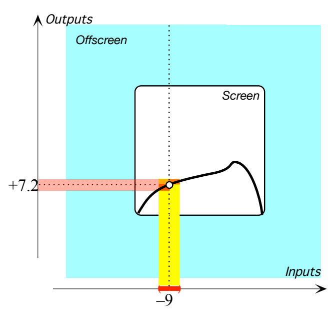


is *height discontinuous* at -9 because the function has a *double jump* at -9 that is:

- even though the outputs for nearby inputs, both inputs *right* of -9 and inputs *left* of -9, are all near $+7.2$,
- the output for -9 itself is $+11.6$.

- A function can be height discontinuous at x_0 because the function has a **gap** at x_0, that is because the function does not return a medium-size output for x_0

EXAMPLE 3.11.
The function

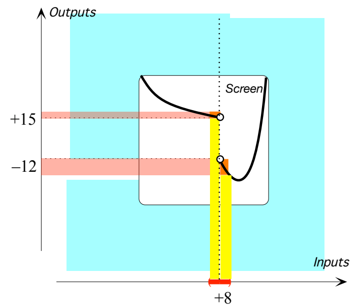


is *height discontinuous* at -9 because the function has a *gap* at -9 that is:

- even though the outputs for nearby inputs, both inputs *right* of -9 and inputs *left* of -9, are all near $+7.2$,
- there is no output for -9 itself.

EXAMPLE 3.12.

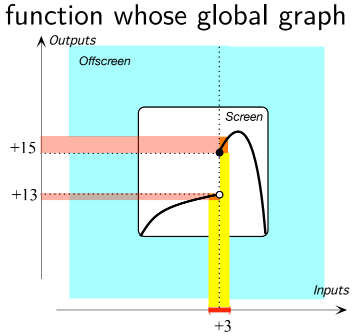
The function



is *height discontinuous* at $+8$ not only because the function has a *jump* at $+8$ but also because the function has a *gap* at $+8$.

EXAMPLE 3.13.

The function whose global graph is



is *height discontinuous* at $+3$ because the global graph has a **jump** at $+3$:

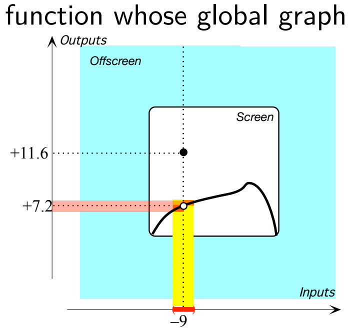
- the outputs for nearby inputs *right* of $+3$ are all near $+15$,

but

- the outputs for nearby Inputs *left* of $+3$ are all near $+13$.

EXAMPLE 3.14.

The function whose global graph is



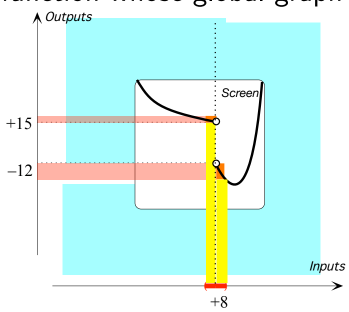
is *height discontinuous* at -9 because the global graph has a **gap** at -9:

- even though the outputs for nearby inputs, both inputs *right* of -9 and inputs *left* of -9, are all near $+7.2$,
- the output for -9 itself is $+11.6$.

EXAMPLE 3.15.

cut-off input
on-off function
transition function
transition

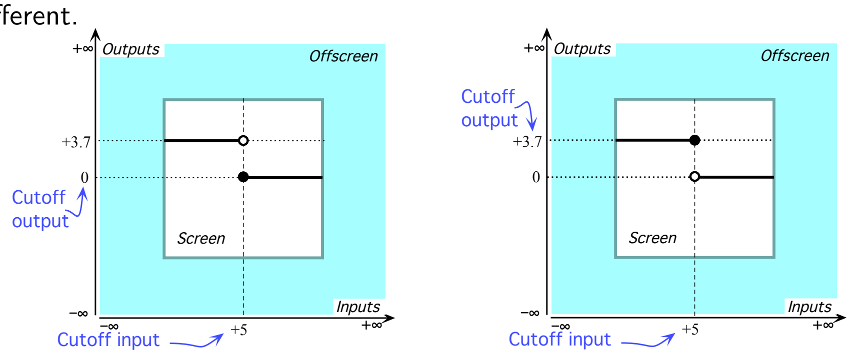
The function whose global graph is



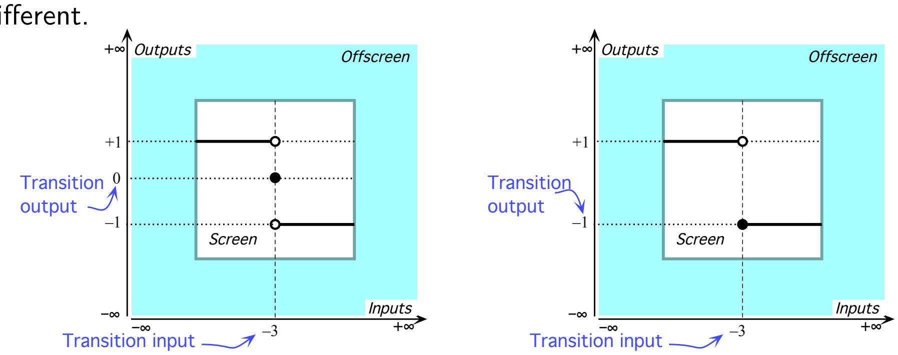
is *height discontinuous* at $+8$ not only because the global graph has a *jump* at $+8$ but also because the global graph has a *gap* at $+8$.

- Actually, height discontinuous functions are quite common in Engineering.

EXAMPLE 3.16. The following **on-off functions** are both *height discontinuous* but are different since the *outputs* for the **cut-off inputs** are different.



EXAMPLE 3.17. The following **transition functions** are both *height discontinuous* but are different since the *outputs* at the **transitions** are different.



- And, finally, there are even functions that are height discontinuous *every-*

Magellan height continuous at limit

where! (https://en.wikipedia.org/wiki/Nowhere_continuous_function)

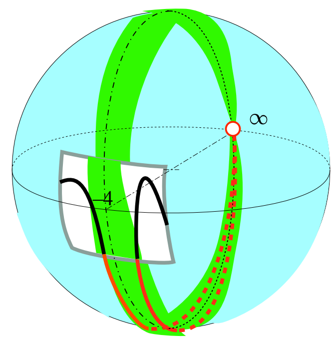
=============OK SO FAR ===============

=======Begin WORK ZONE=======

3. Magellan height-continuity at x_0. A function is **Magellan height continuous at** x_0 when we could remove the height discontinuity at x_0 by overriding or supplementing the global input-output rule with an input-output table involving ∞ as Magellan output.

EXAMPLE 3.18. The function in **??** is *height discontinuous* at -4 because the function has a gap at -4 but *Magellan height continuous* as we could *remove* the gap by *supplementing* the global input-output rule with the input-output table

Input	Output
-4	∞



4. Height-continuity at ∞ The use of nearby inputs instead of the raises a crucial question: Are the outputs for *nearby* inputs *all* near the output *at* the given input?

Any answer, though, will obviously depend on whether or not ∞ is allowed as Magellan input and Magellan output and the reader must be warned that the prevalent stand *in this country* is that ∞ does not exist and that one should use **limits**. (For what limits are, see https://en.wikipedia.org/wiki/Limit_(mathematics).) This for no apparent reason and certainly for none ever given.[1]

As for us, we *will* allow ∞ as Magellan input and Magellan output, an old, tried and true approach. See https://math.stackexchange.com/questions/354319/can_a_function_be_considered_heightcontinuous_if_it_reaches_infinity_at_one_point and, more comprehensively, https://en.wikipedia.org/wiki/Extended_real_number_line.

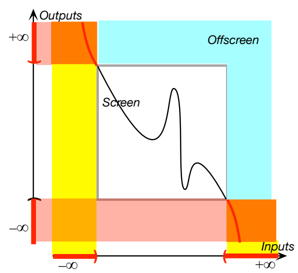
As a backdrop to what we will be doing with Algebraic Functions, we will now show some of the many different possible answers to the above question. For clarity, we will deal with medium-size inputs and medium-size outputs separately from ∞ as Magellan input and Magellan output.

[1]The absolute silence maintained by Educologists in this regard is rather troubling.

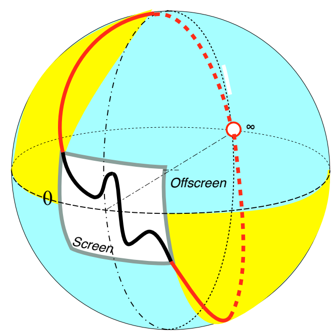
Keep in mind that we use solid dots to represent input-output pairs as opposed to hollow dots which do *not* represent input-output pairs.

5. Magellan height-continuity at ∞. A function is **Magellan height continuous at** ∞ when we could remove the height discontinuity at ∞ by overriding or supplementing the global input-output rule with an input-output table involving ∞ as Magellan input and/or as Magellan output.

EXAMPLE 3.19. The function

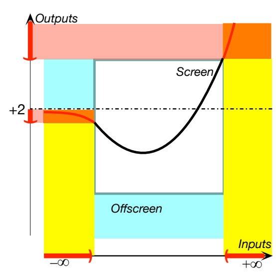


Input	Output
$+\infty$	$-\infty$
$-\infty$	$+\infty$

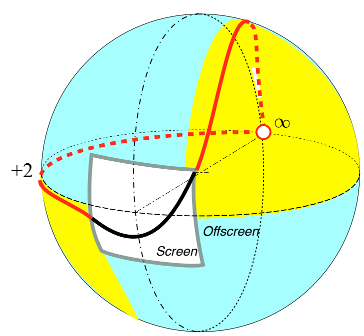


is *height discontinuous* at ∞ but is *Magellan height continuous* since we could remove the height discontinuity with an input-output table involving ∞ as *Magellan input* and *Magellan output*,

EXAMPLE 3.20. The function



Input	Output
$+\infty$	$+\infty$
$-\infty$	-2^-



is *height discontinuous* at ∞ but is *Magellan height continuous* since we could remove the height discontinuity with an input-output table involving ∞ as Magellan input and Magellan output

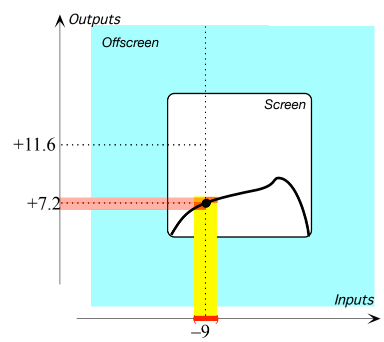
quasi-height continuous at
removable height discontinuity at
remove
override
supplement

6. Quasi height-continuity at x_0. A function is **quasi-height continuous at** x_0 if the height discontinuity could be **removed** by **overriding** or **supplementing** the global input-output rule with an input-output table.

Language 3.1 is the standard term but, for the sake of language consistency, rather than saying that a function *has* (or *does not have*) a removable height discontinuity at x_0, *we* will prefer to say that a function *is* (or *is not*) quasi-height continuous at x_0.

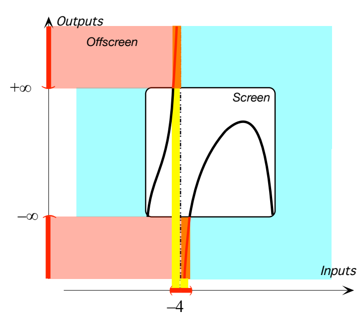
Example 3.21. The function in Example 3.11 is *height discontinuous* at -9 but the height discontinuity could be *removed* by overriding the input-output pair $(-9, +11.6)$ with the input-output table

Input	Output
-9	$+7.2$



A function can be height discontinuous at x_0 because the function has a pole at x_0.

Example 3.22. The function



is *height discontinuous* at -4 because not only does the function have a gap at -4 but the function has a *pole* at -4 that is:

- the outputs for nearby inputs, both inputs *right* of -4 and inputs *left* of -4, are all $large$,

but

- -4 has no medium-size output.

3 Local Extremes

We will often compare the *output at* a given *medium-size* input with the *height near* the given *medium-size* input.

local maximum-height input
$x_{\text{maxi-height}}$
local minimum-height input

1. Local maximum-height input. A **local maximum-height input** is a *medium-size* input whose output is *larger* than the height near the medium-size input. In other words, the output *at* a local maximum-height input is *larger* than the outputs for all nearby inputs.

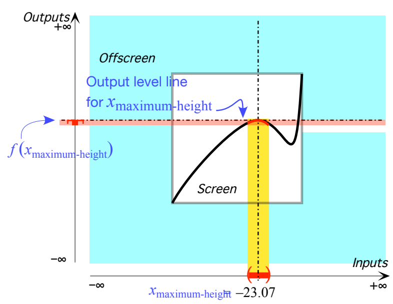
x_0 is al local maximum-height input whenever $f(x_0) > f(x_0 + h)$

We will use $\boldsymbol{x}_{\textbf{max-height}}$ as a name for a local maximum-height input.

Language 3.2 is the usual name for a local maximum-height input but x_{max} tends to suggest that it is the input x that is *maximum* while it is the *output*, $f(x_{\text{max}})$, which is "maximum".

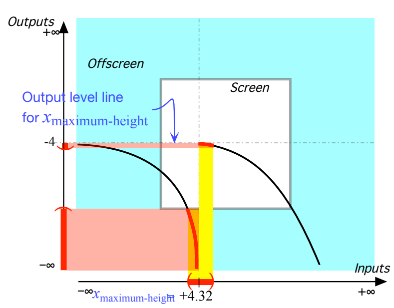
Graphically, the local graph near $x_{\text{max-height}}$ is *below* the output-level line for $x_{\text{max-height}}$.

Example 3.23. The function



has a local *maximum* at -23.07 because the output *at* -23.07 is *larger* than the outputs for *nearby inputs*

Example 3.24. The function



has a local *maximum* at $+4.32$ because the output *at* $+4.32$ is *larger* than the outputs for *nearby inputs*

2. Local minimum-height input. A **local minimum-height input** is a *medium-size* input whose output is *smaller* than the height near the given input. In other words, the output *at* a local minimum-height input is *smaller* than the outputs for all nearby inputs.

x_0 is al local minimum-height input whenever $f(x_0) < f(x_0 + h)$

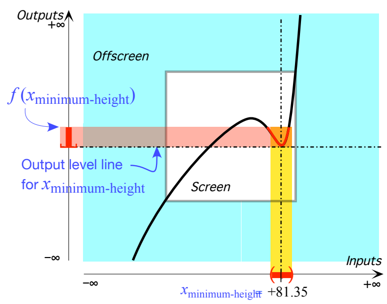
$x_{\text{min-height}}$
local extreme-height input

We will use $\boldsymbol{x}_{\textbf{min-height}}$ as name for a local minimum-height input.

LANGUAGE 3.3 is the usual name for a local minimum-height input but x_{min} tends to suggest that it is the input x that is *minimum* while it is its *output*, $f(x_{\text{min}})$, which is "minimum".

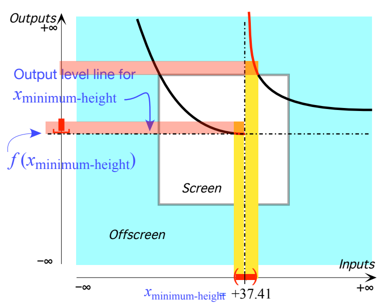
Graphically, the *local graph* near $x_{\text{min-height}}$ is *above* the output-level line for $x_{\text{min-height}}$.

EXAMPLE 3.25. The function



has a local *minimum* at $+81.35$ because the output *at* $+81.35$ is *smaller* than the outputs for *nearby inputs*.

EXAMPLE 3.26. The function



has a local *minimum* at $+37.41$ because the output *at* $+37.41$ is *smaller* than the outputs for *nearby inputs*.

3. Local extreme-height input. Local extreme-height input are *medium-size* inputs which are either a local maximum-height input or a local minimum-height input.

CAUTION 3.1 can only be *medium-size* inputs.

4. Optimization problems. Minimization problems and maximization problems (`https://en.wikipedia.org/wiki/Mathematical_optimization`) local maximum-height input

$x_{\text{maxi-height}}$

as well as min-max problems (https://en.wikipedia.org/wiki/Minimax) are of primary importance in *real life*. So,

- It would be pointless to allow ∞ as a local extreme-height input since it cannot be reached in the *real world*,
- It would be meaningless to allow $+\infty$ as a locally largest output since $+\infty$ is *always* larger than any output or to allow $-\infty$ as a locally smallest output since $-\infty$ is *always* smaller than any output.

5. Local extreme We will often compare the *output at* a given *medium* input with the *height near* the given *medium* input.

6. Local maximum-height input. A **local maximum-height input** is a *medium* input whose output is *larger* than the height near the medium input. In other words, the output *at* a local maximum-height input is *larger* than the outputs for all nearby inputs.

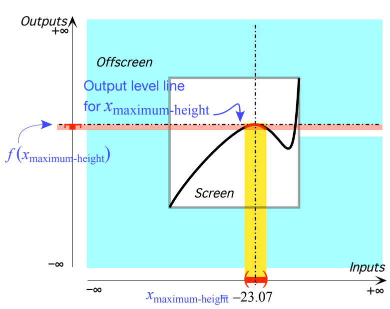
$$x_0 \text{ is al local maximum-height input whenever } f(x_0) > f(x_0 + h)$$

We will use $\boldsymbol{x}_{\textbf{max-height}}$ as a name for a local maximum-height input.

LANGUAGE 3.4 is the usual name for a local maximum-height input but x_{max} tends to suggest that it is the input x that is *maximum* while it is the *output*, $f(x_{\text{max}})$, which is "maximum".

Graphically, the local graph near $x_{\text{max-height}}$ is *below* the output-level line for $x_{\text{max-height}}$.

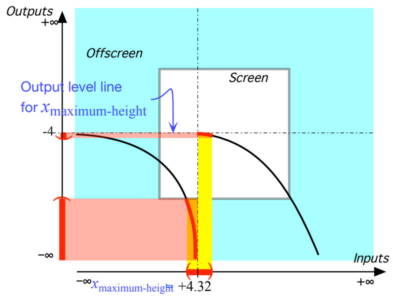
EXAMPLE 3.27. The function has a local *maximum* at -23.07 because the output *at* -23.07 is *larger* than the outputs for *nearby inputs*



local minimum-height input

$x_{\text{min-height}}$

EXAMPLE 3.28. The function



has a local *maximum* at $+4.32$ because the output *at* $+4.32$ is *larger* than the outputs for *nearby inputs*

7. Local minimum-height input. A **local minimum-height input** is a *medium* input whose output is *smaller* than the height near the given input. In other words, the output *at* a local minimum-height input is *smaller* than the outputs for all nearby inputs.

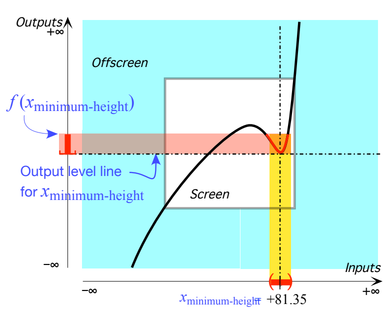
$$x_0 \text{ is al local minimum-height input whenever } f(x_0) < f(x_0 + h)$$

We will use $\boldsymbol{x_{\text{min-height}}}$ as name for a local minimum-height input.

LANGUAGE 3.5 is the usual name for a local minimum-height input but x_{min} tends to suggest that it is the input x that is *minimum* while it is its *output*, $f(x_{\text{min}})$, which is "minimum".

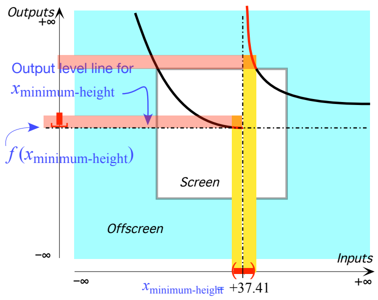
Graphically, the *local graph* near $x_{\text{min-height}}$ is *above* the output-level line for $x_{\text{min-height}}$.

EXAMPLE 3.29. The function



has a local *minimum* at $+81.35$ because the output *at* $+81.35$ is *smaller* than the outputs for *nearby inputs*.

EXAMPLE 3.30. The function



has a local *minimum* at $+37.41$ because the output *at* $+37.41$ is *smaller* than the outputs for *nearby inputs*. local extreme-height input slope-sign

8. Local extreme-height input. **Local extreme-height input** are *medium* inputs which are either a local maximum-height input or a local minimum-height input.

CAUTION 3.2 can only be *medium* inputs.

9. Optimization problems. Minimization problems and maximization problems (https://en.wikipedia.org/wiki/Mathematical_optimization) as well as min-max problems (https://en.wikipedia.org/wiki/Minimax) are of primary importance in *real life*. So,

- It would be pointless to allow ∞ as a local extreme-height input since it cannot be reached in the *real world*,
- It would be meaningless to allow $+\infty$ as a locally largest output since $+\infty$ is *always* larger than any output or to allow $-\infty$ as a locally smallest output since $-\infty$ is *always* smaller than any output.

4 Slope

1. Slope-sign. Inasmuch as, in this text, we will only deal with *qualitative* information we will be mostly interested in the **slope-sign**: .

PROCEDURE 3.3 To get Slope-sign near a given input for a function given by a global graph

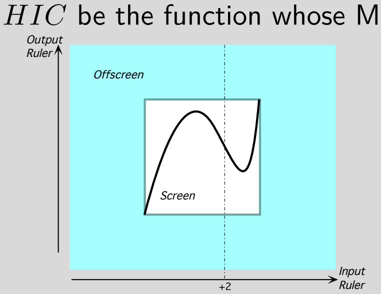
i. Mark the local graph near the given input
ii. Then the slope-sign is:
／ when the local graph looks like ↗ or ↗, that is when the *outputs*

are **increasing** as the inputs are going the way of the input ruler,
╲ when the local graph looks like ╲ or ╲, that is when the *outputs* are **decreasing** as the inputs are going the way of the input ruler.

iii. Code Slope-sign f according to **?? ??** - **??** (**??**)

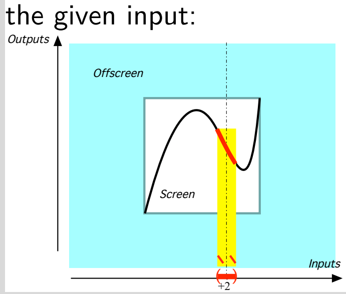
LANGUAGE 3.6 The usual symbols are + Instead of ╱ and − instead of ╲ but, in this text, in order not to overuse + and −, we will use ╱ and ╲.[2]

DEMO 3.3a Let HIC be the function whose Mercator graph is



and let the given input be $+2$. Then to get Slope-sign HIC near $+2$

i. We get the local graph near the given input:



ii. We then get

· The slope sign *left* of $+2$ is ╲

· The slope sign *right* of $+2$ is ╲

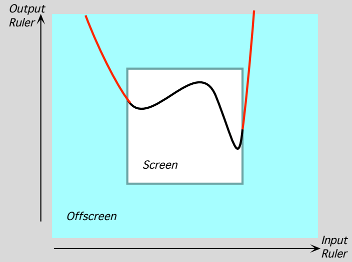
which we code as:

Slope-sign HIC near $+2 = \langle$╲, ╲$\rangle$

DEMO 3.3b Let HIP be the function whose Mercator graph is

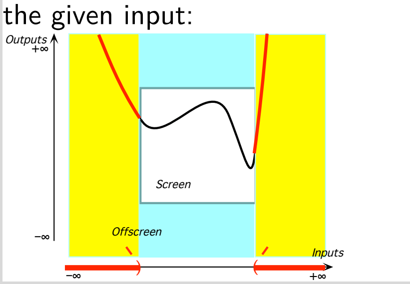
[2]Educologists will surely appreciate "Sign-slope $f =$ ╱ *iff* Sign-heigth $f' = +$".

slope-size



and let the given input be ∞. Then to get Slope sign HIP near ∞

i. We get the local graph near the given input:



ii. We then get that:

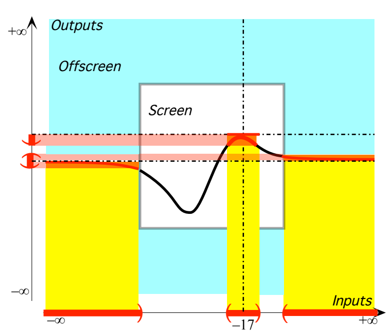
- The slope sign *left* of ∞, that is near $+\infty$, is $\diagup$
- The slope sign *right* of ∞, that is near $-\infty$, is $\diagdown$

which we code as:

Slope-sign HIP near $\infty = \langle \diagup, \diagdown \rangle$

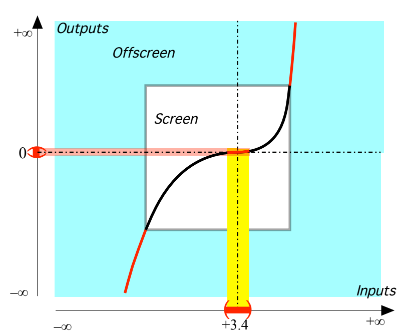
2. Slope-size In this text, we will not deal with **slope-size** other than in the case of a 0**-slope input** that is an input, be it x_0 or ∞, near which slope-size is *small*. This is because 0-slope inputs will be extremely important in *global analysis* as finding 0-slope inputs comes up all the time in direct "applications" to the real world:

EXAMPLE 3.31. The function



has both -17 and ∞ as 0-slope inputs

EXAMPLE 3.32. The function



Only $+3.4$ is a 0-slope input.

EndWORKzone - EndWORKzone - EndWORKzone - EndWORKzone - EndWORKzone

kink
concavity
concavity-size
concavity-sign

5 Slope-continuity

1. Tangent. The *first* degree of smoothness is for the *slope* not to have any abrupt change.

to be height continuous, that is, to borrow a word from plumbing, we don't want the curve to have any **kink**. More precisely, we don't want any input x_0 for which there is a "jump in slope" from one side of x_0 to the other side of x_0. In other words, we don't want any input x_0 for which the slope on one side differs from the slope on the other side by some medium-size number.

6 Concavity

1. Concavity-sign Inasmuch as, in this text, we will be only interested in *qualitative analysis* we will not deal with the **concavity-size** but only with the **concavity-sign**:

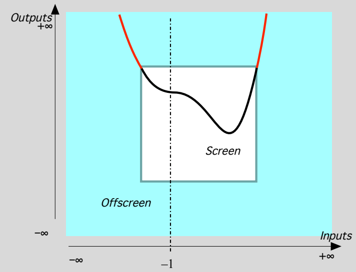
PROCEDURE 3.4 To get Concavity-sign near a given input for a function given by a *global graph*

i. Mark the local graph near the given input
ii. Then the concavity-sign is:
$\cup$ when the local graph is *bending up* like ◟ or ◞,
$\cap$ when the local graph is *bending down* like ◜ or ◝.
iii. Code Slope-sign f according to **??** **??** - **??** (**??**)

LANGUAGE 3.7 The usual symbols are $+$ Instead of $\cup$ and $-$ instead of $\cap$ but, in this text, in order not to overuse $+$ and $-$, we will use $\cup$ and $\cap$.[3]

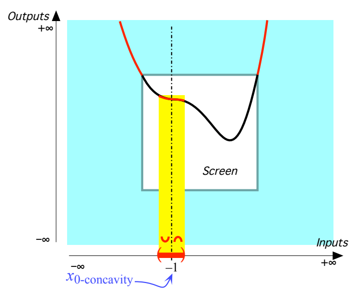
DEMO 3.4 Let KIP be the function whose Mercator graph is

[3] Educologists will surely appreciate "Sign-concavitye $f = \cup$ *iff* Sign-heigth $f'' = +$".



and let the given input be -1. Then to get Concavity sign KIP near -1

i. We get the local graph near the given input:



ii. We then get that:
- The concavity sign *left* of -1, is $\cup$
- The concavity sign *right* of -1, is $\cap$

which we code as:

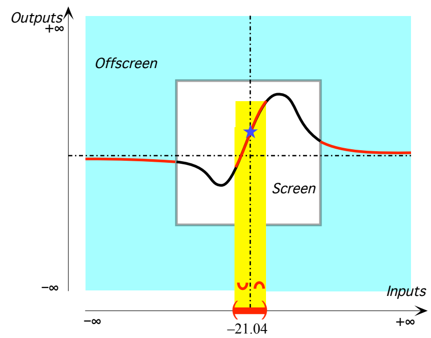
Concavity Sign KIP near $-1 = \langle \cup, \cap \rangle$

2. 0-concavity input. Given a function f, the inputs whose Concavity-size is 0 will be particularly important in *global analysis*:

A *medium* input x_0 is a 0-**concavity input** if inputs that are near x_0 have *small concavity*. We will use $x_{0\text{-concavity}}$ to refer to 0-concavity inputs.

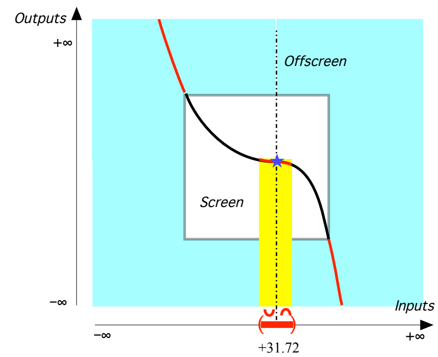
extremity

EXAMPLE 3.33. Given the function whose Mercator graph is



$x_{0-concavity} = -21.04$

EXAMPLE 3.34. Given the function whose Mercator graph is

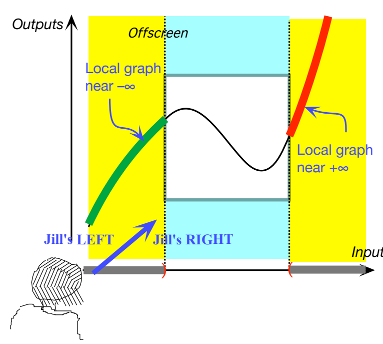


$x_{0-concavity} = +31.72$

Under AGREEMENT 1.1 - (Page 65), with only a Mercator view of the global graph, there is of course no way we can get the whole local graph near ∞ and we will have to content ourselves with just the **extremities** of the local graph near ∞. However, since we cannot face ∞ and can only face the screen, we have to keep in mind Subsection 2.5 - Poles (Page 108)) so that

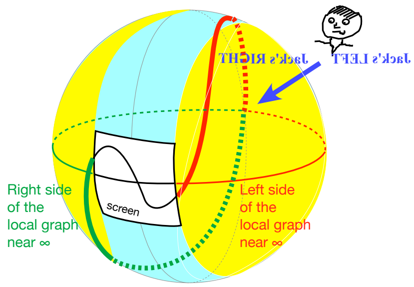
- The extremity of the local graph near $+\infty$ (*left of* ∞) is to *our right*,
- The extremity of the local graph near $-\infty$ (*right of* ∞) is to *our left*.

EXAMPLE 3.35.

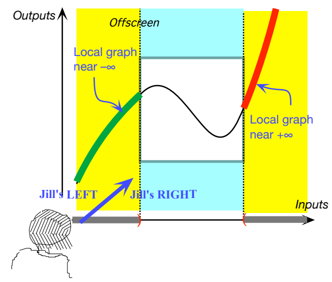


Jill is facing the *screen* so she can only see the *extremities* of the local graph near ∞ and she must keep in mind Subsection 2.5 - Poles (Page 108)) so that the local graph near $+\infty$ (to *her right*) is *left* of ∞ and the local graph near $-\infty$ (to *her left*) is *right* of ∞.

EXAMPLE 3.36.



When facing the *screen*, though, Jill can only see the *extremities* of the local graph near ∞ and she must keep in mind that the local graph near $+\infty$ (*left* of ∞) is to Jill's *right* and the local graph near $-\infty$ (*right* of ∞) is to Jill's *left*.



When facing the *screen*, though, Jill can only see the *extremities* of the local graph near ∞. As a result, the local graph near $+\infty$ (*left* of ∞) is to Jill's *right* and the local graph near $-\infty$ (*right* of ∞) is to Jill's *left*.

that is the largest error that will not change the qualitative information we are looking for. The largest permissible error, which is the equivalent of a tolerance, will turn out to be easy to determine.

We can see from Chapter 3 that the reason could not possibly give us a global graph is that, if a plot point may tell us where the global graph "is at", a plot point certainly cannot tell us anything about where the global graph "goes from there". And, since the latter is precisely what local graphs do with slope and concavity, we are now in a position to:

==================================

Something wrong with references here

1. *Describe* how to interpolate local graphs into a global graph. This corresponds to the second of the **??** **??** - **??** (**??**)

2. Discuss questions about interpolating local graphs which correspond to the other two **??** **??** - **??** (**??**)

i. How will we know near which inputs to get the local graphs?

ii. After we have interpolated the local graphs, how will we know if the curve we got *is* the global graph?

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**
OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

7 Concavity-continuity

1. Osculating circle. The *second* degree of smoothness is for the *concavity* not to have any abrupt change.

to be height continuous but this is much harder to represent because it is hard to judge by just looking how much a curve is bending.

=======Begin WORK ZONE=======

=======End WORK ZONE========

2. Dealing with poles. The difficulty here stems only from whether or not it is "permisible" to use ∞ as a given input and/or as an output.

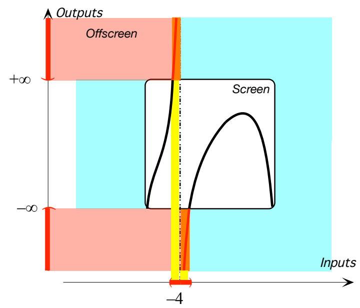
Even though, because **??** **??** - **??** (**??**) (**??** **??** - **??** (**??**)), **??** **??** - **??** (**??**,) *we do* use ∞ as a (Magellan) input and as a (Magellan) output because, as explained in **??** (**??**), we will *only* declare nearby inputs. (Which will shed much light on the local behavior of functions, in particular on the question of height continuity.)

However, the reader ought to be aware that many mathematicians *in this country*, for reasons never stated, flatly refuse to use nearby inputs with their students.

Another reason *we do* is because Magellan views are a very nice basis on which to discuss the local behavior of functions for inputs near ∞ and when outputs are near ∞. In particular, we can see that disheight continuiities caused by poles can be removed using ∞ as a Magellan output.

When ∞ as is not permissible as Magellan input and/or Magellan output, many functions are height discontinuous

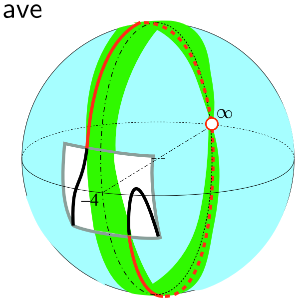
Example 3.37. The height discontinuity at -4 of the function in **??** whose Mercator graph is



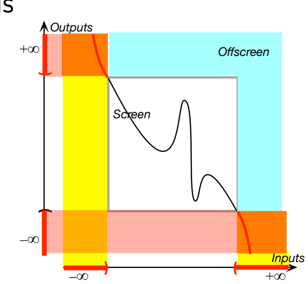
can be *removed* by *supplementing* the global input-output rule with the input-output table:

Input	Output
-4	∞

If we imagine the Mercator graph compactified into a Magellan graph, we have



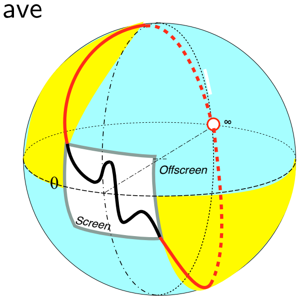
EXAMPLE 3.38. The height discontinuity at ∞ of the function BIB in **??** whose Mercator graph is



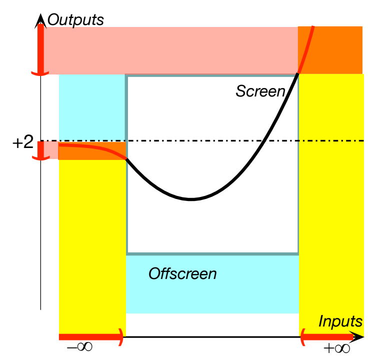
can be *removed* by *supplementing* the global input-output rule with the input-output table:

Input	Output
∞	∞

If we imagine the Mercator graph compactified into a Magellan graph, we have

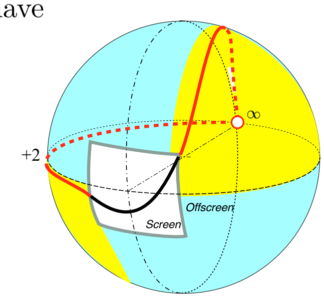


EXAMPLE 3.39. The function whose the global graph in *Mercator view* is



is *height discontinuous* at ∞ not only because the global graph has a *gap* at ∞ since **??** **??** - **??** (**??**) but also because the global graph has a *jump* at ∞.

If we imagine the Mercator view *compactified* into a Magellan view, we have



3. At ∞ The matter here revolves around whether or not ∞ should be allowed as a given input. *We* did but,

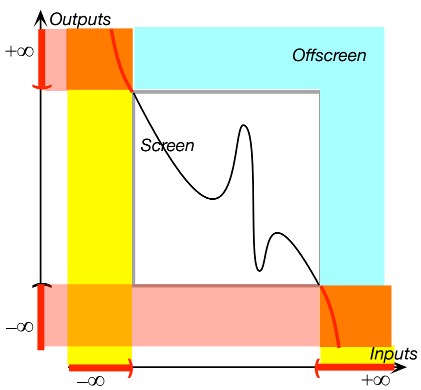
Also, in this section, for a reason which we will explain after we are done, we will have to deal separately with the case when the given input is x_0 and the case when the given input is ∞.

In accordance with **??**, we should say that all functions are height discontinuous at ∞ since the outputs for inputs near ∞ cannot be near the output for ∞ for the very good reason that we cannot use ∞ as input to begin with.

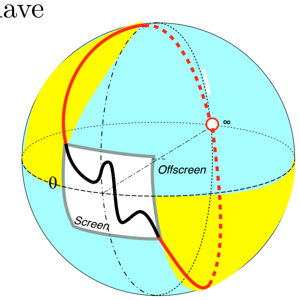
LANGUAGE 3.8 At ∞, things are a bit sticky:

- With a Magellan view, we can *see* if a function is height continuous at ∞ or not.
- Technically, though, to talk of height continuity at ∞ requires being able to take computational precautions not worth taking here but many teachers feel uneasy dealing with height continuity at ∞ without taking these precautions. So, we will not discuss height continuity at ∞ in this text.

EXAMPLE 3.40. The function whose global graph in Mercator view is

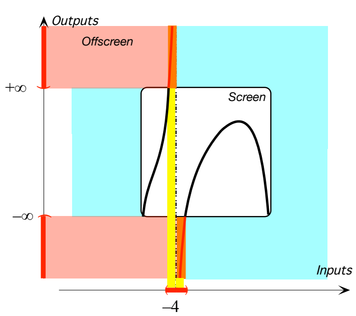


If we imagine the Mercator view *compactified* into a Magellan view, we have



is *height discontinuous* at ∞ because, even though the outputs of inputs near ∞ are all $large$,the global graph has a gap at ∞ since **??**.

EXAMPLE 3.41. The function



is *height discontinuous* at -4 because the global graph has a **pole** at -4:

- the outputs for nearby inputs, both inputs *right* of -4 and inputs *left* of -4, are all $large$,

but, since **??**,

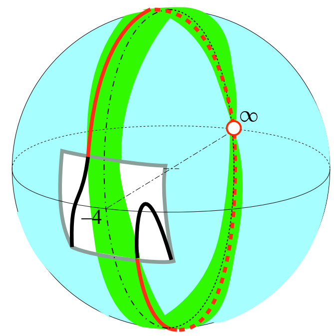
- -4 itself has no output.

Magellan height continuous at

4. Magellan height-continuity at a pole x_0. *We* will say that a function is **Magellan height continuous at** x_0 when we can remove the height discontinuity at x_0 supplementing the offscreen graph with an input-output table involving ∞ as Magellan output.

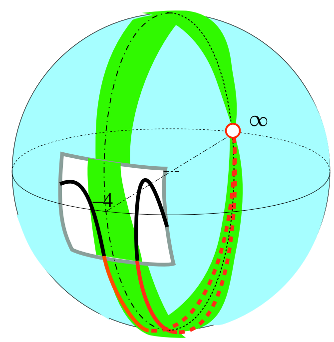
Example 3.42. The function in **??** is *height discontinuous* at -4 because the function has a gap at -4 but *Magellan height continuous* as we could *remove* the gap by *supplementing* the global input-output rule with the input-output table

Input	Output
-4	∞



Example 3.43. The function in **??** is *height discontinuous* at -4 because the function has a gap at -4 but *Magellan height continuous* as we could *remove* the gap by *supplementing* the global input-output rule with the input-output table

Input	Output
-4	∞



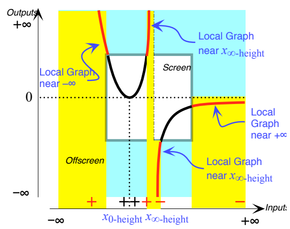
==========OK SO FAR ===========
=======End HOLDING =======
=======Begin WORK ZONE=======

8 Feature Sign-Change Inputs

We will often need to find *medium* inputs such that the outputs for nearby inputs left of x_0 and the outputs for nearby inputs right of x_0 have given feature-signs.

1. height sign-change input An input is a **height sign-change input**whenever height sign $= \langle +, - \rangle$ or $\langle -, + \rangle$. We will use $x_{\textbf{height sign-change}}$to refer to a *medium* height sign-change input.

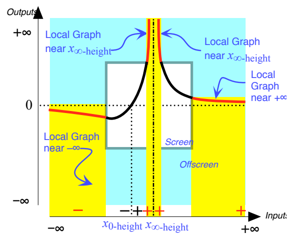
EXAMPLE 3.44.
Let f be the function given by the global graph



Then,

- $x_{0\text{-height}}$ *is not* a height sign-change input,
- $x_{\infty\text{-height}}$ *is* a height sign-change input.
- ∞ *is* a height sign-change input.

EXAMPLE 3.45.
Let f be the function given by the global graph

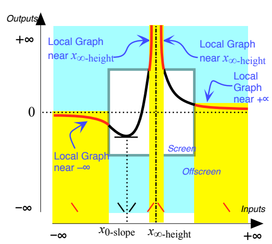


Then,

- $x_{0\text{-height}}$ *is* a height sign-change input,
- $x_{\infty\text{-height}}$ *is not* a height sign-change input,
- ∞ *is* a height sign-change input.

2. Slope sign-change input An input is a **Slope sign-change input** whenever Slope sign $= \langle \diagup, \diagdown \rangle$ or $\langle \diagdown, \diagup \rangle$. We will use $x_{\textbf{Slope sign-change}}$ to refer to a Slope sign-change input.

EXAMPLE 3.46.
Let f be the function given by the global graph

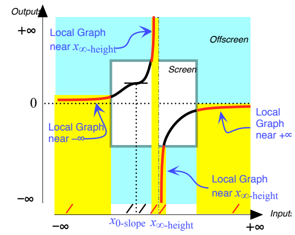


Then,

- $x_{0\text{-slope}}$ *is* a Slope sign-change input,
- $x_{\infty\text{-height}}$ *is* a Slope sign-change input,
- ∞ *is not* a Slope sign-change input.

EXAMPLE 3.47.

Let f be the function given by the global graph



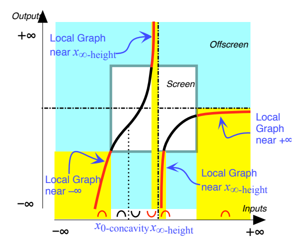
Then,

- $x_{0\text{-slope}}$ *is not* a Slope sign-change input,
- $x_{\infty\text{-slope}}$ *is not* a Slope sign-change input,
- ∞ *is not* a Slope sign-change input.

3. Concavity sign-change input An input is a **Concavity sign-change input**whenever Concavity sign $= \langle \cup, \cap \rangle$ or $\langle \cap, \cup \rangle$. We will use $x_{\textbf{Concavity sign-change}}$to refer to a Concavity sign-change input.

EXAMPLE 3.48.

Let f be the function given by the global graph

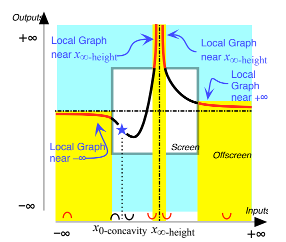


Then,

- $x_{0\text{-concavity}}$ *is* a Concavity sign-change input,
- $x_{\infty\text{-height}}$ *is* a Concavity sign-change input.
- ∞ *is not* a Concavity sign-change input.

EXAMPLE 3.49.

Let f be the function given by the global graph



Then,

- $x_{0\text{-concavity}}$ *is* a Concavity sign-change input,
- $x_{\infty\text{-height}}$ *is not* a Concavity sign-change input,
- ∞ *is* a Concavity sign-change input.

One case where the picture gets a bit complicated is when the *output* point is ∞, that is when the *input* point is a *pole*

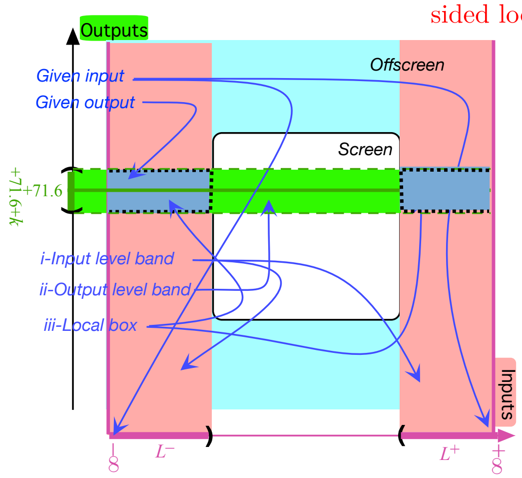
The two other cases where the picture gets a bit complicated are when the *input* point is ∞, whether the *output* point is a number y_0 or ∞.

EXAMPLE 3.50. Local box for the input-output pair $\left(\infty, +71.6 \right)$

sided local graph box

i. We get the input level *band* for ∞
ii. We get the output level *band* for $+71,6$
iii. We box the intersection of the input level bands for ∞ and $+71.6$

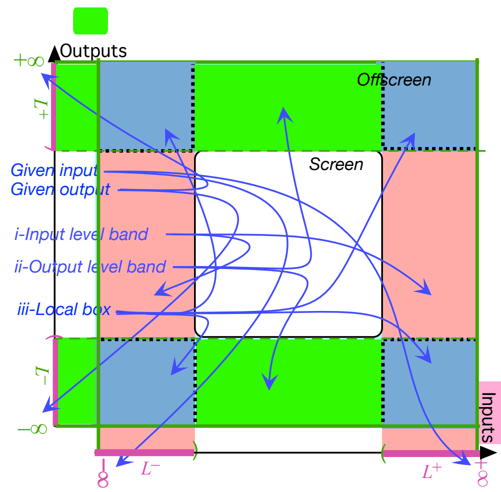
What appears to be two boxes are actually parts of *one* box. This is because we are using the Mercator view. In a Magellan view they would appear as a single box.



EXAMPLE 3.51. Local box for the input-output pair (∞, ∞)

i. We get the input level *band* for ∞
ii. We get the output level *band* for ∞
iii. We box the intersection of the input level bands for ∞ and ∞

What appears to be four boxes are actually parts of *one* box. This is because we are using the Mercator view. In a Magellan view they would appear as a single box.



Actually, we will very often want to keep the two sides of. separate and the **sided local graph box** will then consist of two smaller rectangles, one on each side of the input level line. To get a sided local graph place then,

PROCEDURE 3.5
i. Mark a *neighborhood* of the input on the input ruler,
ii. Draw the *input level band*,
iii. Mark a *neighborhood* of the output on the output ruler,

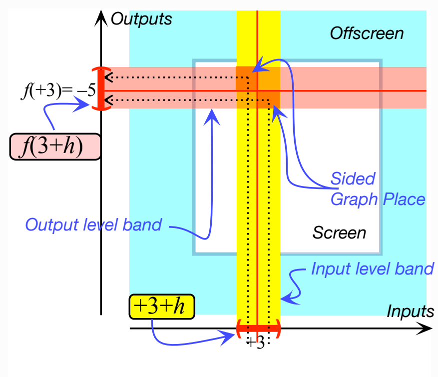
iv. Draw the *output level band*,
v. Mark which side of the input neighborhood is linked to which side of the output neighborhood,
vi. The place for the given input - output pair is at the intersection of the corresponding *sides* of the level bands.

DEMO 3.5 Get the sided place for ($+3$, -5) given that:

- $+3^{-} \longrightarrow -5^{+}$
- $+3^{+} \longrightarrow -5^{-}$

i. We mark a *neighborhood of* $+3$ on the *input ruler*,
ii. We draw the *input level band* through the *neighborhood* of $+3$,
iii. We mark a *neighborhood of* -5 on the *output ruler*,
iv. We draw the *output level band* through the *neighborhood* of -5,
v. Mark:

- left of $+3 \to$ above -5
- right of $+3 \to$ below -5



vi. The *sided graph box* for ($+3$, -5) is at the intersection of the corresponding *sides* of the level bands.

Quite a long way away from "just plugging" numbers into the global input-output rule and joining smoothly the plot dots". But that will be graphing that makes sense.

We are now going to sketch the way we will graph functions given by I-O rules which we will illusttrate with an extended EXAMPLE.

The big missing piece is that we will only be able to get the local frames and will not be able to really justify the local graphs until Chapter 3.

The general idea will be to

4. Offscreen graph. Local graph(s) near the control input(s)

i. Local graph near ∞. We saw in EXAMPLE 1.15 that $\left(L \,,\, -2 \oplus [\ldots] \right)$

ii. Local graph(s) near the pole(s), if any.

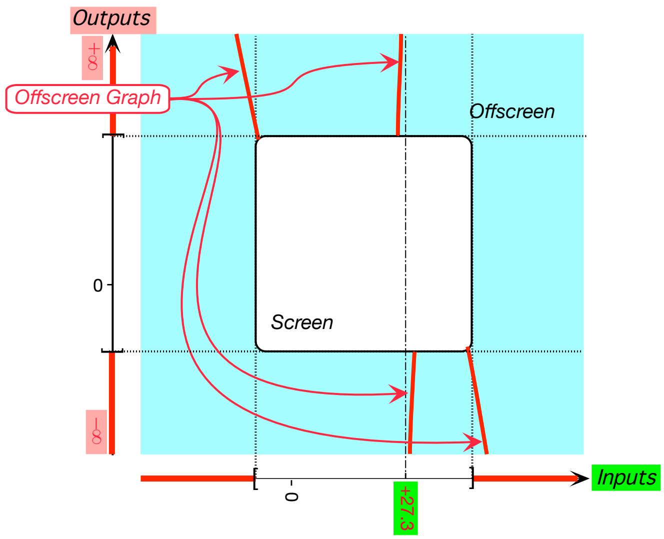
We saw in EXAMPLE 1.12 that -7 is a pole for the function $JILL$.

We saw in EXAMPLE 1.14 that $\left(-7 \oplus h,\, L + [\ldots] \right)$

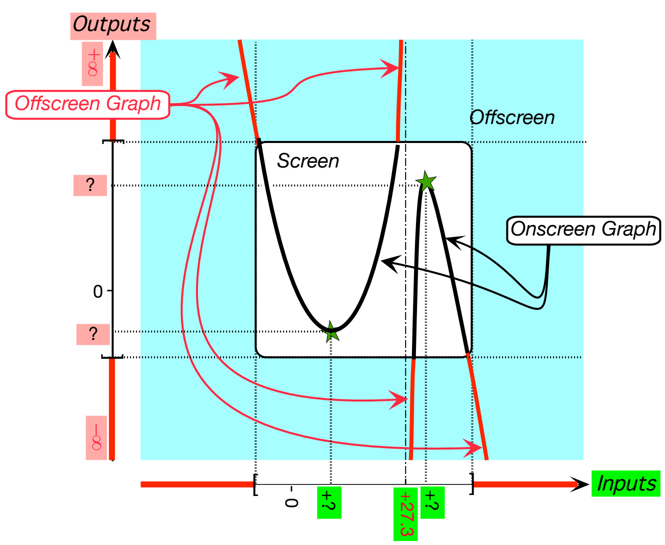
iii. Offscreen graph.

Very roughly speaking! The smooth talk will begin in the next chapter.

EXAMPLE 3.52. Consider the offscreen graph of the function IAN in EXAMPLE 1.11:



Joining smoothly this offscreen graph on-screen gives something like:



which is pretty much like IAN's actual on-screen graph and even shows IAN's 'essential' features, namely that:

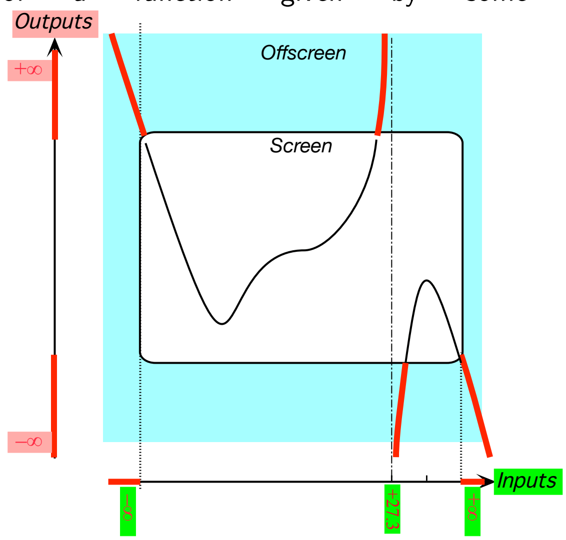
- IAN has *a* 'minimum point', (But of course does *not* show what the input-output pair is.)

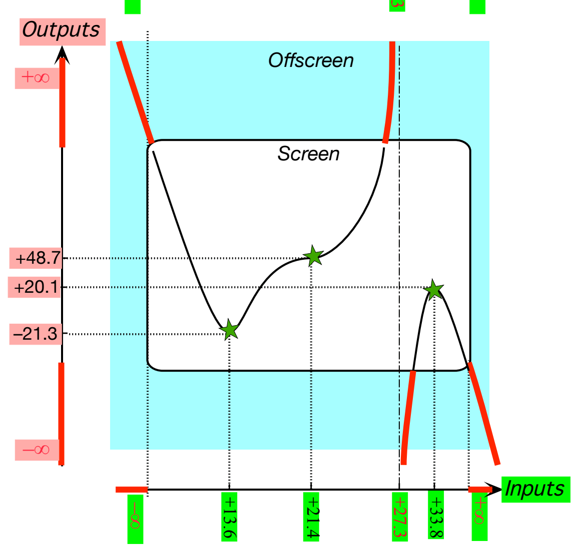
- IAN has *a* ‘maximum point’, (But of course does *not* show what the input-output pair is.)

but does *not* show that IAN has an ‘inflection point’.

==============OK SO FAR ===============

EXAMPLE 3.53. Say the following is the global graph of a function given by some I-O rule:





We can *see* from the picture that the given function has:

- What we will call a ‘pole’: ($+27.3$, ∞).

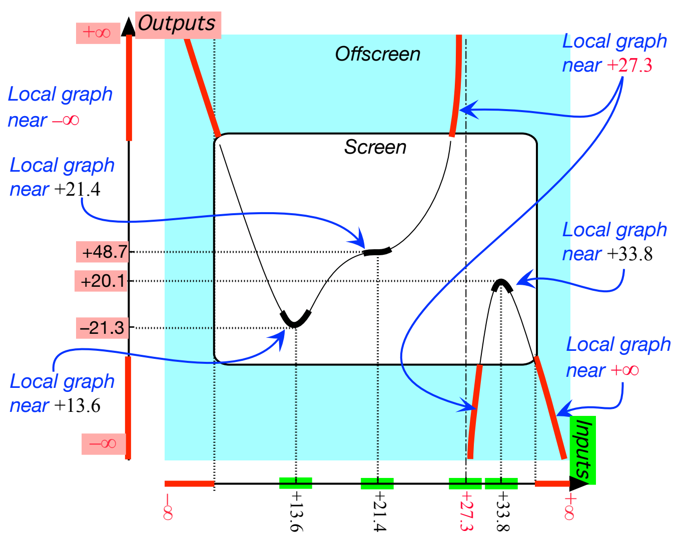
Sneak preview!
join smoothly

and

- What we will call a 'minimum point': ($+13.6$, -21.3),
- What we will call an 'inflection point': ($+21.4$, $+48.7$),
- What we will call a 'maximum point': ($+33.8$, $+20.1$),

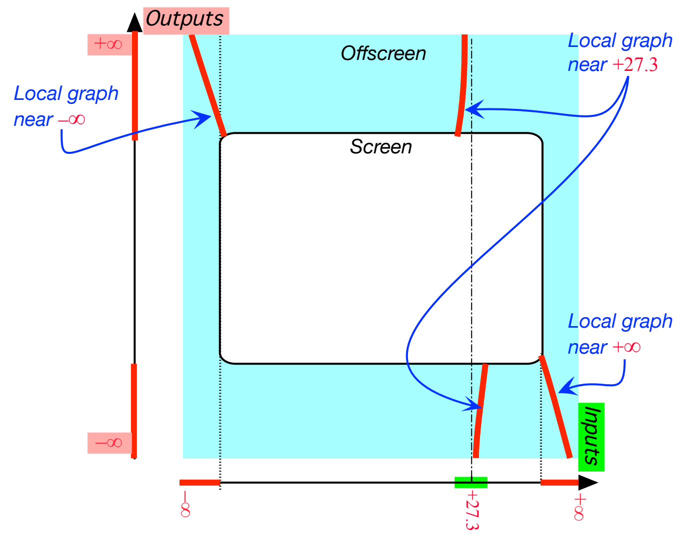
Most *important!*

EXAMPLE 3.54. In EXAMPLE 1.13, the local graphs are:

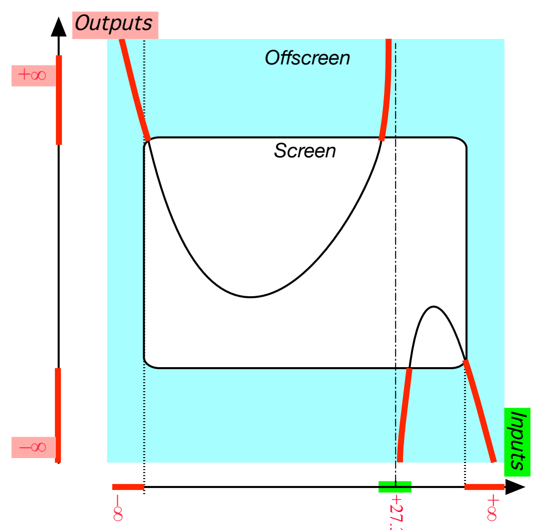


Conversely, our approach to getting the global graph of a function given by an I-O rule will be to use the I-O rule to get the poles of the given function, if any, and then **join smoothly** the local graphs near the pole(s), if any, and near ∞.

EXAMPLE 3.55. To get the global graph in EXAMPLE 1.14 we first get the control local graphs:



which we then join smoothly:



Notice, though, that we while we *did* recover the 'existence' of a 'maximum point' right of $+27.3$ and the 'existence'of a 'minimum point' left of $+27.3$, we did *not* recover the 'existence' of an 'inflection point'.

5. Sided local frame.

We obtain the procedure to get a sided local graph frame just by thickening **??** (**??**):

PROCEDURE 3.6

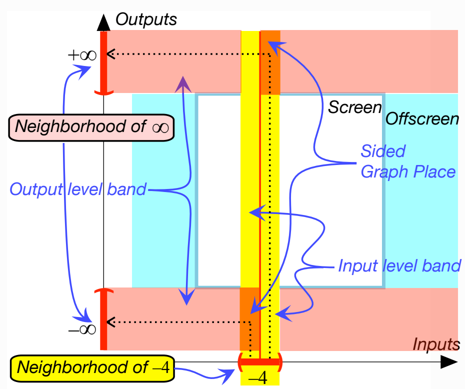
i. Mark a *neighborhood* of the input on the input ruler,
ii. Draw the *input level band*,
iii. Mark a *neighborhood* of the output on the output ruler,
iv. Draw the *output level band*,
v. Mark which side of the input neighborhood is linked to which side of the output neighborhood,
vi. The local graph box for the given input - output pair is at the intersection of the corresponding *sides* of the level bands.

DEMO 3.6 Get the sided local graph frame for $(-4, \infty)$ given that:

- $-4^- \longrightarrow -\infty$
- $-4^+ \longrightarrow +\infty$

i. We mark a *neighborhood of* -4 on the *input ruler*,
ii. We draw the *input level band* through the *neighborhood* of -4,
iii. We mark a *neighborhood of* ∞ on the *output ruler*,
iv. We draw the *output level band* through the *neighborhood* of ∞,
v. Mark:

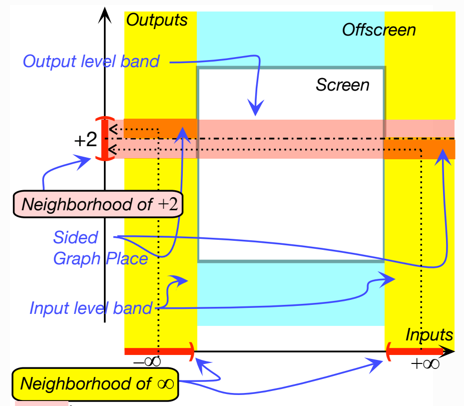
- left of -4 → near $-\infty$
- right of -4 → near



vi. The *sided graph box* for $(-4, \infty)$ is at the intersection of the corresponding *sides* of the level bands.

DEMO 3.7 Get the sided local graph frame for $(\infty, +2)$ given that:

- $-\infty \longrightarrow +2^+$
- $+\infty \longrightarrow +2^-$



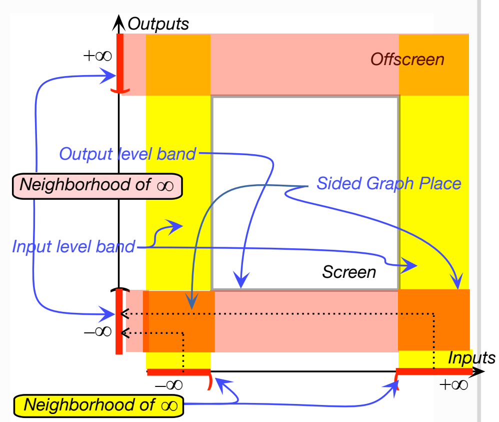
i. We mark a *neighborhood of* ∞ on the *input ruler*,
ii. We draw the *input level band* through the *neighborhood* of ∞,
iii. We mark a *neighborhood of* $+2$ on the *output ruler*,
iv. We draw the *output level band* through the *neighborhood* of $+2$,
v. Mark:

- $-\infty \to +2^+$
- $+\infty \to +2^-$

vi. The *sided graph box* for (∞, $+2$) is at the intersection of the corresponding *sides* of the level bands.

DEMO 3.8 Get the sided local graph frame for (∞, ∞) given that:

- $-\infty \longrightarrow -\infty$
- $+\infty \longrightarrow -\infty$



i. We mark a *neighborhood of* ∞ on the *input ruler*,
ii. We draw the *input level band* through the *neighborhood* of ∞,
iii. We mark a *neighborhood of* ∞ on the *output ruler*,
iv. We draw the *output level band* through the *neighborhood* of ∞,
v. Mark:

- $-\infty \to -\infty$
- $+\infty \to -\infty$

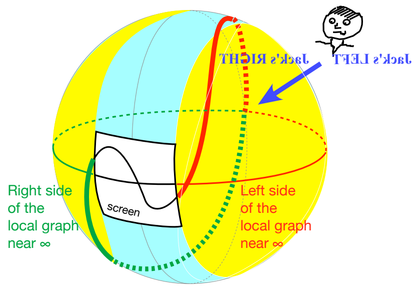
vi. The *sided graph box* for (∞, ∞) is at the intersection of the corresponding *sides* of the level bands.

With a Magellan view of the global graph, we proceed pretty much as in **??** and once we imagine facing ∞, we can *see* which side is which.

==========================

EXAMPLE 3.56.

essential



Jack is facing ∞ so the local graph near $+\infty$ which is to *his left* is *left* of ∞ and the local graph near $-\infty$ which is *to* his right *is right* of ∞.

9 *Essential* Feature-Sign Changes Inputs

1. Essential sign-change input A feature sign-change input is **essential** whenever its **existence**is **forced**by the offscreen graph. So, given the offscreen graph of a function, in order

PROCEDURE 3.7 To establish the existence of essential feature sign change inputs in a inbetween curve

i. For each piece of the inbetween curve, check the feature sign at both end of the piece.

- If the feature signs at the two ends of the piece are *opposite*, there *has* to be a feature sign change input for that piece.
- If the feature signs at the two ends of the piece are the *same*, there does *not* have to be a feature sign change input for that piece.

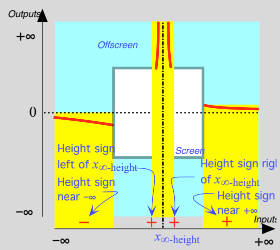
ii. For each ∞ height input, if any, check the feature sign on either side of the ∞ height input:

- If the feature signs on the two sides of the ∞ height input are *opposite*, the ∞ height input *is* a feature sign change input.
- If the feature signs on the two sides of the ∞ height input are the *same*, the ∞ height input *is not* a feature sign change input..

iii. Check the feature sign on the two sides of ∞

- If the feature signs on the two sides of ∞ are *opposite*, ∞ *is* a feature sign change input.
- If the feature signs on the two sides of ∞ are the *same*, ∞ *is not* a feature sign change input..

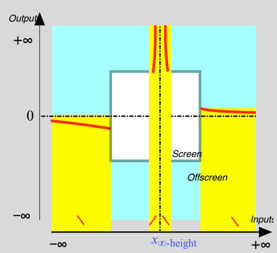
DEMO 3.9a



To establish the existence of Height-sign change inputs

- Since the Height signs near $-\infty$ and left of $x_{\infty\text{-height}}$ are *opposite* there *is* an essential Height sign-change between $-\infty$ and $x_{\infty\text{-height}}$.
- Since the Height signs right of $x_{\infty\text{-height}}$ and near $+\infty$ are *the same* there *is no* essential Height sign-change between $x_{\infty\text{-height}}$ and $+\infty$.

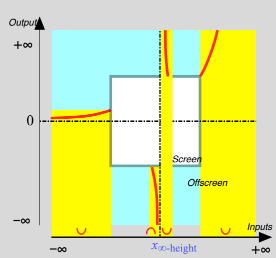
DEMO 3.9b



To establish the existence of Slope-sign change inputs

- Since the Slope signs near $-\infty$ and left of $x_{\infty\text{-height}}$ are *opposite* there *is* an essential Slope sign-change between $-\infty$ and $x_{\infty\text{-height}}$.
- Since the Slope signs right of $x_{\infty\text{-height}}$ and near $+\infty$ are *the same* there *is no* essential Slope sign-change between $x_{\infty\text{-height}}$ and $+\infty$.

DEMO 3.9c

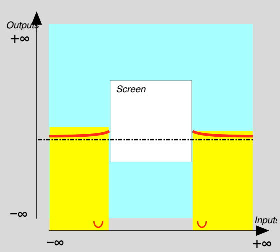


To establish the existence of Concavity-sign change inputs

- Since the Concavity signs near $-\infty$ and left of $x_{\infty\text{-height}}$ are *opposite* there *is* an essential Concavity sign-change between $-\infty$ and $x_{\infty\text{-height}}$.
- Since the Concavity signs right of $x_{\infty\text{-height}}$ and near $-\infty$ are *the same* there *is no* essential Concavity sign-change between $x_{\infty\text{-height}}$ and $+\infty$.

2. more complicated However, things can get a bit more complicated.

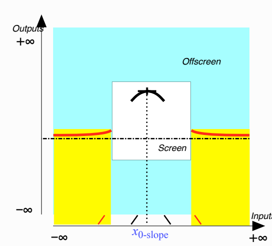
DEMO 3.9d



To establish the existence of Concavity-sign change inputs

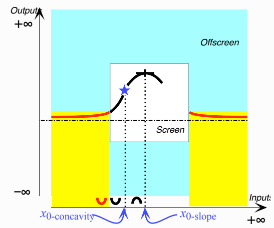
- Since the concavity-sign at the transitions from $-\infty$ is $\cup$ and the concavity-sign at the transition to $+\infty$ is also $\cup$, one might be tempted to say that there is no essential concavity sign-change input.
- However, attempting a smooth interpolation shows that things are a bit more complicated than would at first appear.

i. Since the slope-signs at the transition *from* $-\infty$ is $\diagup$ and the slope-sign at the transition *to* $+\infty$ is $\diagdown$ there has to be an essential Slope sign-change input near which Concavity sign $= \langle \cap, \cap \rangle$

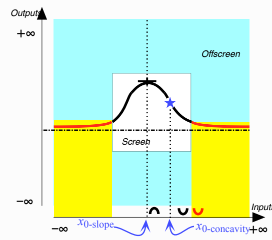


essential␣local␣extreme-height␣input

ii. Since the concavity-signs near $-\infty$ and *left* of $x_{0\text{-slope}}$ are *opposite*, there is an essential Concavity sign-change input between $-\infty$ and $x_{0\text{-slope}}$.

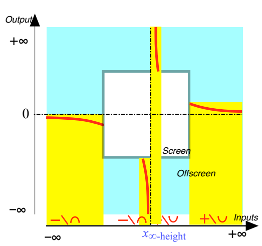


iii. Since the concavity-signs *right* of $x_{0\text{-slope}}$ and near $+\infty$ are *opposite*, there is an essential Concavity sign-change input between $x_{0\text{-slope}}$ and $+\infty$.

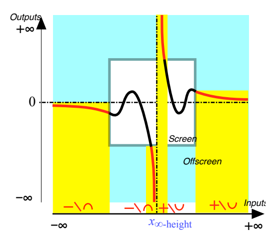


3. non-essential That there is no *essential* feature sign-change input does not mean that there could not actually be a *non-essential* feature sign-change input.

EXAMPLE 3.57.
Let f be the function whose offscreen graph is



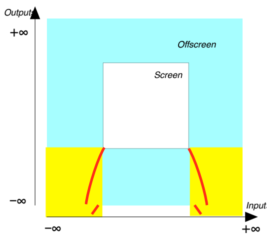
- There is no *essential* Height sign-change input, no *essential* Slope sign-change input, and no *essential* Concavity sign-change input.
- However, the actual medium-size graph could very well be:



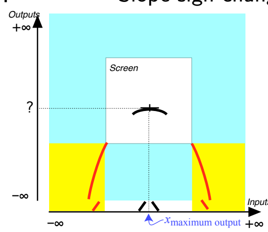
4. Essential Extreme-Height Inputs An extreme-height input is an **essential local extreme-height input** if the existence of the local extreme-height input is forced by the offscreen graph in the sense that *any* smooth interpolation *must* have a local extreme-height input.

EXAMPLE 3.58.

Let f be a function whose offscreen graph is

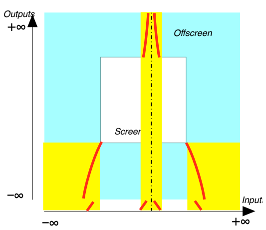


Then,
i. Since the Slope signs near $-\infty$ and $+\infty$ are *opposite* there *is* an essential Slope sign-change between $-\infty$ and $+\infty$.
ii. Since the Height of $x_{\text{Slope sign-change}}$ is not infinite, the slope near $x_{\text{Slope sign-change}}$ must be 0



iii. $x_{\text{0-slope}}$ is a local essential Maximum-Height input.

EXAMPLE 3.59.
Let f be a function whose offscreen graph is



Then,
i. Since the Slope signs near $-\infty$ and near $+\infty$ are *opposite* there *is* an essential Slope sign-change between $-\infty$ and $+\infty$.
ii. But since there is an ∞-height input, the Height near $x_{slopesign-change}$ is infinite and there is no essential local maximum height input.

5. Non-essential Features While, as we have just seen, the *offscreen graph* may force the existence of certain feature-sign changes in the *onscreen graph*, there are still many other smooth interpolations of the *offscreen graph* that are not forced by the onscreen graph.

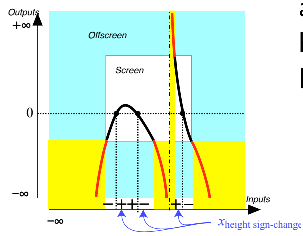
EXAMPLE 3.60. The moon has an influence on what happens on earth—for instance the tides—yet the phases of the moon do not seem to have an influence on the growth of lettuce (see http://www.almanac.com/content/farming-moon) or even on the mood of the math instructor.

We will say that a global feature is **non-essential**if it is *not* forced by the offscreen graph.

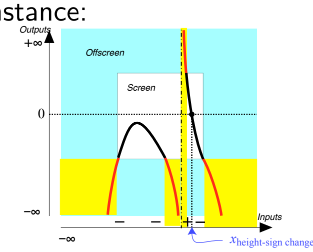
1. As we saw above, feature sign-change inputs can be non-essential.

bump
wiggle

EXAMPLE 3.61.
Let f be a function whose graph is



Then,
i. The two Height sign-change inputs left of $x_{\infty\text{-height}}$ are non-essential because if the 0-output level line were higher, there would be no Height sign-change input. For instance:

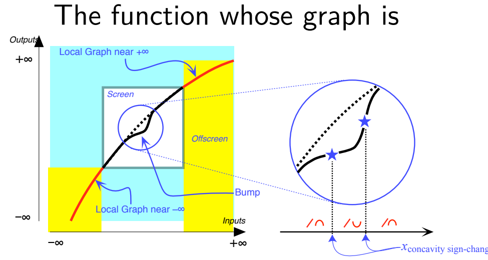


ii. The Height sign-change input right of $x_{\infty\text{-height}}$ is essential because, no matter where the 0-output level line might be, the inbetween curve has to cross it.

2. There other non-essential features:

- A *smooth* function can have a **bump** in which the slope does not change sign but the concavity changes sign twice.

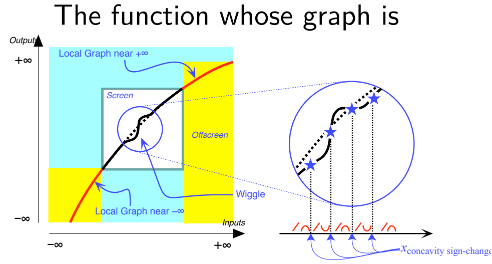
EXAMPLE 3.62. The function whose graph is



has a *bump*.

- A *smooth* function can also have a **wiggle**, that is a pair of bumps in opposite directions with the slope keeping the same sign throughout but with *three* inputs where the concavity changes sign.

EXAMPLE 3.63. The function whose graph is

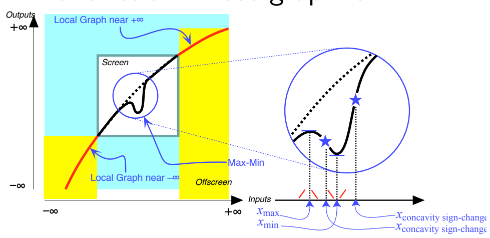


has a *wiggle*.

max-min fluctuation
min-max fluctuation
essential onscreen graph
join smoothly

- A *smooth* function can also have a **max-min fluctuation** or a **min-max fluctuation** that is a sort of "extreme wiggle" which consists of a pair of *extremum-heights inputs* in opposite directions. In other words, a fluctuation involves:
  - *two* inputs where the *slope* changes sign
  - *two* inputs where the *concavity* changes sign

EXAMPLE 3.64. The function whose graph is



has a *max-min fluctuation*.

However, as we will see in Chapter 4 - Input-Output Rules (Page 197), in Mathematics, functions are not usually given by a curve but are given "mathematically" and the investigation of how a function given "mathematically" behaves *cannot* be based on the function's global graph which, in any case, is usually not necessarily simple to get as we will discuss in Section 4 - Outputs *Near* A Given *Number* (Page 204).

But, while finding the global graph of a function given "mathematically" is not stricly necessary to *understand* how the given function behaves, the global graph of a function given "mathematically" *can* be a very great help to *see* the way the given function behaves.

So, in order to explain how we will get the global graph of a function given "mathematically" we will have to proceed by stages using functions given by a curve.

We begin by outlining the PROCEDURE which we will follow in Chapter 4 - Input-Output Rules (Page 197).

i. The first step in getting the global graph of a function given "mathematically" will be to get the local graphs near the control points, that is near ∞ and near the poles, if any.

ii. The second step in getting the global graph of a function given "mathematically" will be to get the offscreen graph.

iii. The third step in getting the global graph of a function given "mathematically" will be to get the **essential onscreen graph** by **joining smoothly**

essential graph
join smoothly
essential on-screen graph
existence
proximate on-screen graph

the offscreen graph across the screen.

6. The essential onscreen graph. Thus, the first step in getting the global graph of a function given by an I-O rule will be to get the **essential graph**, that is the onscreen graph forced by the offscreen graph, in other words, the onscreen graph as we would see it from very far away.

PROCEDURE 3.8 To get the essential graph of a function given by a global input-output rule

i. Get the offscreen graph, that is,
- **a.** Get the local graph near ∞,
- **b.** Get the local graph near the pole(s), if any,

ii. Join smoothly the offscreen graph across the screen

Get the offscreen graph from the local graphs near the control inputs namely near ∞ and near the pole(s) if any,

Then get the ***essential* on-screen graph** by

And just because something is far away doesn't mean it's of no interest*:* "Many ancient civilizations collected astronomical information in a systematic manner through observation.*"* *(https://en.wikipedia.org/wiki/History_of_science.)*

The *essential* on-screen graph will already provide information about the **existence** of *essential* behavior change inputs *on-screen*—but *not* about their location.

However, there might be behavioral changes too small to see from far away, so get the **proximate on-screen graph** by:

a. locating the non-essential behavioral change inputs, if any,

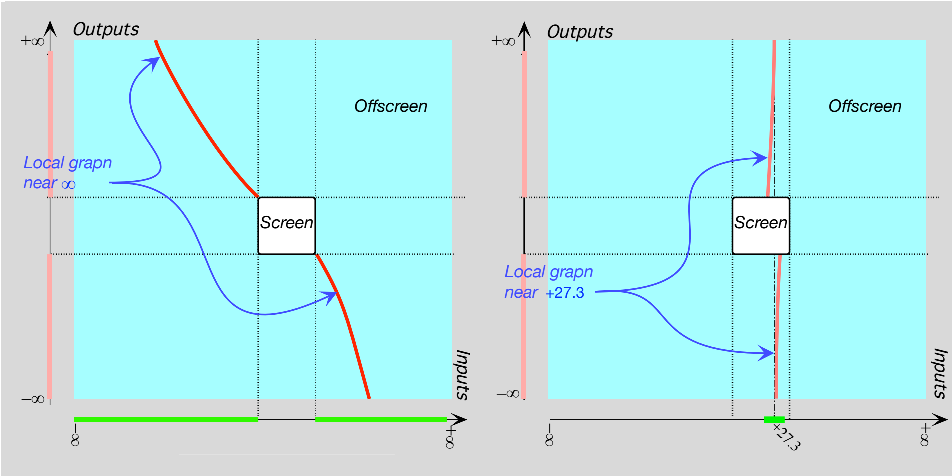
b. getting the local graphs near these non-essential behavioral change inputs

Actually makes sense doesn't it?

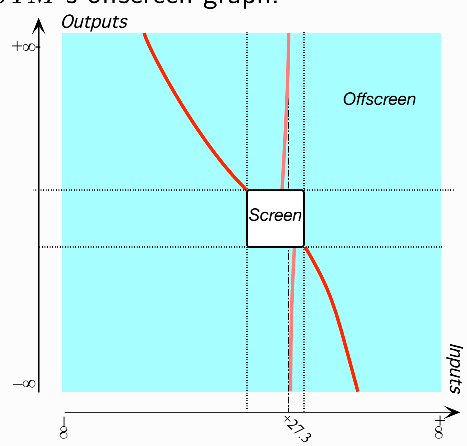
c. Joining smoothly *all* local graphs,

and then progressively zero in:

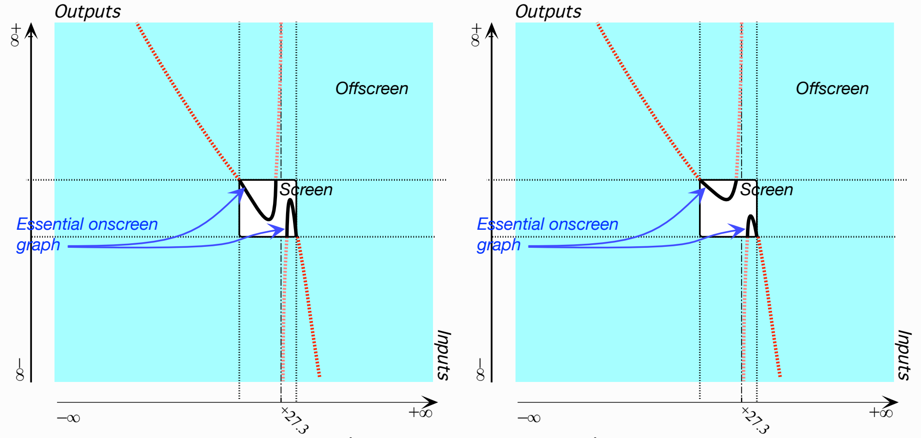
DEMO 3.10 Suppose that we found that the function JIM has a pole at $+27.3$ and that the local graphs near $+27.3$ and ∞ look like



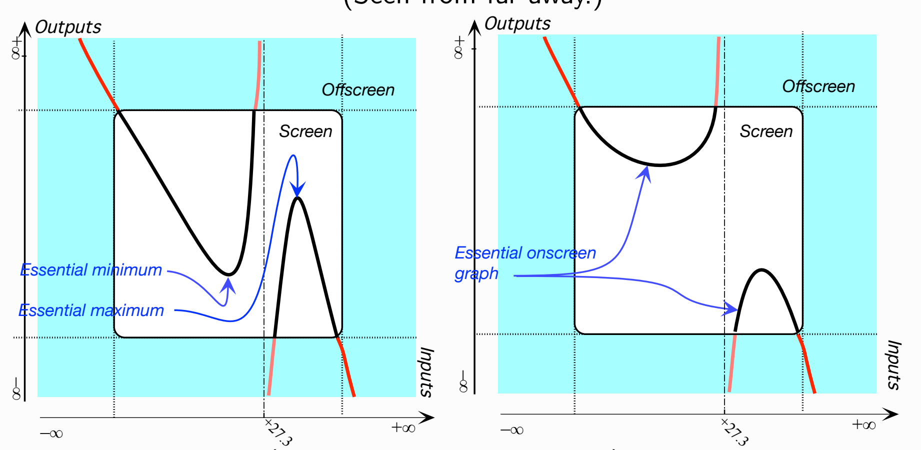
I. We then have JIM's offscreen graph:



II. Then, the essential on-screen graph of JIM would look something like either one of:

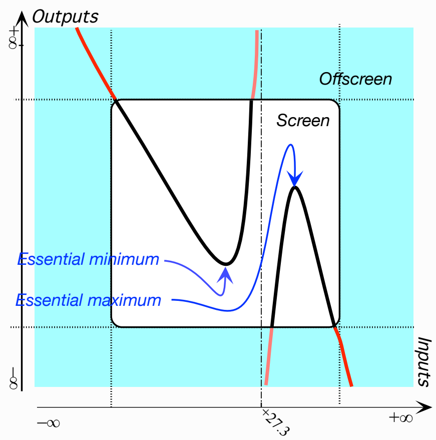


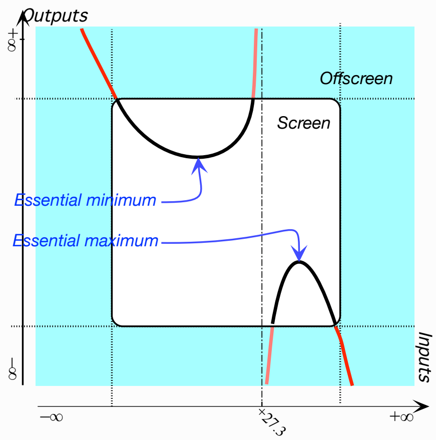
(Seen from far away.)



(Seen from a short distance.)

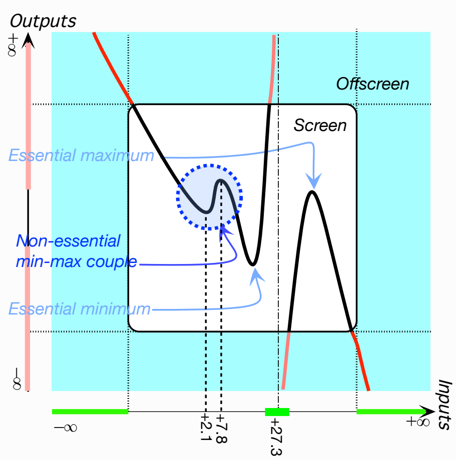
Under the assumption that the function is continuous, the *essential* on-screen graph of JIM already shows, the existence of an *essential* minimum and of an *essential* maximum:

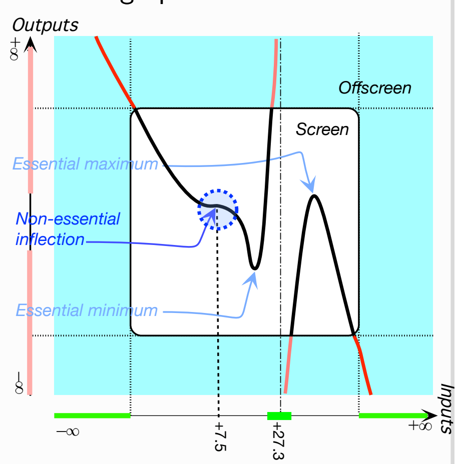




However, there might be behavioral changes that we are too far to see.

III. Supposing, for instance, that we were to locate from the I-O rule a maximum at $+7.8$ coupled with a minimum at $+2.1$, or a non-essential inflection at $+7.5$ then the proximate graph would be:





If we were to locate *no* non-essential behavioral change input, then of course the proximate graph would just be the essential on-screen graph

10 EmptyA

1. EmptyAa

2. EmptyAb

smooth

3. EmptyAc

11 EmptyB

1. EmptyBa

2. EmptyBb

3. EmptyBc Roughly, **smoothness** extends to slope and concavity the requirements that height continuity made on the height namely that there should be no abrupt differences in slope and concavity. This is quite another thing though:

- In the case of height continuity, we need to look at what happens *at* the given input and then to what happens *near* the given input but only to see if there is a jump and not even when there is a gap at x_0.
- In the case of slope and concavity, on the other hand, even with local graphs, neither slope nor concavity makes sense *AT* the given input and what matters is only what happens *NEAR* the given input.

CAUTION 3.3 Most unfortunately, the *usual* mathematical concept of smoothness implies height continuity which is not the way we think of smoothness in the real world.

For that matter, educologists well know that, in order to define smoothness at x_0 in the usual way one needs room in which to have a limit.

EXAMPLE 3.65. A PVC sewer pipe is usually perceived as being "smooth" regardless of whether or not it is solid or perforated and a smoothly bending copper pipe doesn't stop being so if and when it develops a pinhole.

So, in this text and in trying to *represent* smoothness, *we* will go by $f(x_0 + h)$ and not pay any attention to $f(x_0)$.

https://en.wikipedia.org/wiki/Smoothness.

https://en.wikipedia.org/wiki/Analytic_function

https://en.wikipedia.org/wiki/Singularity_(mathematics)

https://en.wikipedia.org/wiki/Nowhere_heightcontinuous_function

https://en.wikipedia.org/wiki/Weierstrass_function

https://en.wikipedia.org/wiki/Fractal_curve

12 Start

characterize
local feature
global featur

1. substart The purpose of this chapter is to introduce and discuss a number of 'features' that a function may or may not have when considering certain inputs.

An important matter will be to **characterize** inputs with regards to functions.

We will begin with **local features**, that is 'features' that a function may or may not have when considering inputs in a neighborhood of a given point, be the point a given number x_0 or ∞.

We will then continue with **global feature**s, that is 'features' that a function may or may not have when considering *ALL* inputs.

Part II

Calculatable Functions

While Functions Given By Data (Part I, Page 63) are often used in the *experimental* sciences, Functions Given By Data do not lend themselves to the calculations necessary for an *understanding* of how the function works.

So, both in engineering and in the sciences, functions are mostly given in ways that allow the output to be calculated and this **Part II** deals with the first and simplest way to do so.

explicit

Chapter 4

Input-Output Rules

We now introduce the first of the two ways to give a calculatable function. describe, description

and leave the second way to ?? (?? ??, ??).

1 Giving Functions Explicitly

1. Global Input-Output Rules To give a function f **explicitly** is to give:

i. A global variable for the input numbers to be used—*RBC* will normally employ x,

ii. A Global expression in terms of x for the output numbers to be computed in terms of the input numbers:

Definition 4.1 A **global Input-Output rule** (**global I-O rule** for short) provides a global expression in terms of x for computing $f(x)$ in terms of x:

pointwise format

$$\underbrace{\underbrace{x}_{\text{Input}} \xrightarrow{\ f\ } \underbrace{\underbrace{f(x)}_{\text{Output}} = \text{Global expression in terms of } x}_{\text{Global output rule}}}_{\text{Global input-output rule}}$$

$$\underbrace{underbracedstuff}_{underlabel}$$

EXAMPLE 4.1. To give the function called $\mathcal{JILL}$ explicitly, we give the global variable x and the global expression $\frac{(-4\odot x)\oplus+7}{+2\odot(x\oplus+7)}$ so that $\mathcal{JILL}$ is given by:

$$\underbrace{\underbrace{x}_{\text{Input}} \xrightarrow{\ \mathcal{JILL}\ } \underbrace{\underbrace{\mathcal{JILL}(x)}_{\text{Output}} = \frac{(-4\odot x)\oplus+7}{+2\odot(x\oplus+7)}}_{\text{Global output rule}}}_{\text{Global input-output rule}}$$

2. Format Input-Output pairs. With functions given by a global Input-Output rule, *RBC* will employ the following **pointwise formats** for Input - Output pairs.

DEFINITION ?? (Restated) ??
When the input is either a regular input or a pole, *RBC* will employ:

- The data pair format for *plotting*:
  $\left\langle x_0, y_0 \right\rangle$ when x_0 is a regular input
  or
  $\left\langle x_0, \infty \right\rangle$ when x_0 is a pole
- The function format for *computing*:
  $f\left(x_0 \right\rangle = y_0$ when x_0 is a regular input
  or
  $f\left(x_0 \right\rangle = \infty$ when x_0 is a pole
- The arrow format for *visualizing*:
  $x_0 \xrightarrow{f} y_0$ when x_0 is a regular input
  or
  $x_0 \xrightarrow{f} \infty$ when x_0 is a pole
- The full arrow format for *anything*:
  $x_0 \xrightarrow{f} f\left(x_0 \right) = y_0$ when x_0 is a regular input
  or
  $x_0 \xrightarrow{f} f\left(x_0 \right) = \infty$ when x_0 is a pole

2 Output *AT* A Given Number.

Even though, as we argued in Subsection 9.2 - Nearby numbers (Page 45), evaluating a global expression *AT* a given number is to ignore the real world, we *will* occasionally need, if only for plotting purposes, to get the outputs for inputs *AT* given numbers and the PROCEDURE *RBC* will employ will essentiallybe just the same as PROCEDURE 0.1 - Get an individual expression from a global expression (Page 14):

PROCEDURE 4.1 To get the output, if any, returned for an input *AT* a given number x_0 by a function f given by a global I-O rule

$$x \xrightarrow{f} f(x) = \text{global expression in terms of } x,$$

In standard CALCULUS *texts the two steps, declaration and execution, are often conflated into a single step but we will keep the two steps separate.*

a. Declare that the input is *AT* the given number x_0 by writing the declaration

$$\Big|_{x \leftarrow x_0} \quad \text{read as} \quad \Big|_{x \text{ to be replaced by } x_0}$$

to the right of the global I-O rule:

$$x \xrightarrow{f} f(x) = \text{global expression in terms of } x \Big|_{x \leftarrow x_0}$$

b. Execute the declaration by replacing in the global I-O rule every occurence of the global variable x by the given number x_0 to get the individual expression for $f\left(x_0\right)$, that is for the output returned by f *AT* x_0:

$$x_0 \xrightarrow{f} f\left(x_0\right) = \text{individual expression for } f\left(x_0\right)$$

c. Try to compute the individual expression for $f\left(x_0\right)$, that is try to carry out the computations to find the value of $f\left(x_0\right)$, that is the number, if any, that the function returns as output when the input is *AT* the given number x_0.

d. Characterize the input number with regards to the function.

e. Format the input-output pair according to **??** **??** - **??** (**??**)

Demo 4.1a To get the output *AT* -5 returned by the function $\mathcal{JILL}$ given by the global I-O rule $x \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x) = \dfrac{(-4\odot x)\oplus +7}{+2\odot(x\oplus +7)}$

a. We declare that the input is *AT* -5, that is we write the declaration

$$\Big|_{x \leftarrow -5}$$

to the right of the global input-output rule:

$$x \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x) = \frac{(-4\odot x)\oplus +7}{+2\odot(x\oplus +7)}\Bigg|_{x \leftarrow -5}$$

b. We execute the declaration, that is we replace every occurence of x in the global input-output rule by -5 to get the individual expression for $\mathcal{JILL}(-5)$ that is for the output of $\mathcal{JILL}$ *at* -5:

$$-5 \xrightarrow{\mathcal{JILL}} \mathcal{JILL}\big(-5\big) = \frac{\big(-4 \odot -5\big) \oplus +7}{+2 \odot \big(-5 \oplus +7\big)}$$

c. We try to compute the individual expression for $\mathcal{JILL}(-5)$: :

$$= \frac{(+20) \oplus +7}{+2 \odot (+2)}$$
$$= \frac{+20 \oplus +7}{+2 \odot +2}$$
$$= \frac{+27}{+4}$$
$$= +6.75$$

d. We characterize the input -5 with regards to the function $\mathcal{JILL}$: Since the function $\mathcal{JILL}$ returns a *number*, $+6.75$, for the input -5, the input -5 is a *regular* input for the function $\mathcal{JILL}$.

e. We format the input-output pair, that is:

- For *plotting*, we use the data pair format $\big(-5, +6.75\big)$
- For *computing*, we use the function format $\mathcal{JILL}(-5) = +6.75$
- For *visualizing*, we use the arrow format $-5 \xrightarrow{\mathcal{JILL}} +6.75$
- For *anything*, we can use the full arrow format $-5 \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(-5) = +6.75$

OK, don't worry too much about the algebra: the idea for this Demo *and for the* Examples *that will follow is only to impress you with the power and the scope of* Procedure *2.3. So, for the time being, the most important for you is to develop an appreciation of jusr the way* Procedure *2.3 works.*

However, an individual expression need not always compute to a number and in particular, as we saw in ?? ?? - ?? (??), when having to divide a non-zero number by 0, we can write the result as being ∞ and say that the input is a ?? (?? ??, ??).

Demo 4.1b To get the output returned for -7 by $\mathcal{JILL}$ given by the global input-output rule $x \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x) = \frac{(-4\odot x)\oplus+7}{+2\odot(x\oplus+7)}$

a. We declare that the input is *AT* -7, that is we write the declaration

$$\Big|_{x \leftarrow -7}$$

to the right of the global input-output rule:

$$x \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x) = \left.\frac{(-4 \odot x) \oplus +7}{+2 \odot (x \oplus +7)}\right|_{x \leftarrow -7}$$

b. We execute the declaration, that is we replace every occurence of x in the global input-output rule by -7 to get the individual expression for $\mathcal{JILL}(-7)$ that is for the output of $\mathcal{JILL}$ *AT* -7:

$$-7 \xrightarrow{\mathcal{JILL}} \mathcal{JILL}\big(-7\big) = \frac{\big(-4 \odot -7\big) \oplus +7}{+2 \odot \big(-7 \oplus +7\big)}$$

c. We try to compute the individual expression for $\mathcal{JILL}(-7)$, that is we perform all the operations:

$$= \frac{(+28) \oplus +7}{+2 \odot (0)}$$
$$= \frac{+28 \oplus +7}{+2 \odot 0}$$
$$= \frac{+35}{0}$$
$$= \infty$$

d. We characterize the input -7 with regards to the function $\mathcal{JILL}$: Since the function $\mathcal{JILL}$ returns ∞ for the input -7 instead of returning a *number*, the input -7 is a *pole* of the function $\mathcal{JILL}$.

e. We format the input-output pair, that is:

- For *plotting*, we use the data pair format $\big(-7, \infty\big)$
- For *computing*, we use the function format $\mathcal{JILL}(-7) = \infty$
- For *visualizing*, we use the arrow format $-7 \xrightarrow{\mathcal{JILL}} \infty$
- For *anything*, we can use the full arrow format $-7 \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(-5) = \infty$

3 A Few Words of Caution Though.

input versus input number

When a function is given by a global I-O rule instead of by a global graph, though, we will have to be very careful before we use **??** because

In Subsection 4.3 - Local frame (Page 130) we discussed how to get a local graph when the function is given by a smooth curve. When the function is given by an I-O rule, though, we start out with no global graph, though, and getting a local graph is much more complicated and will require the knowledge of the global graphs of 'power functions'.

Since $x_0 \oplus h$ is a thickening of x_0, it is most tempting and natural to think of $f\left(x_0 \oplus h\right)$ as a thickening of $f\left(x_0\right)$ but, even though it is "often" the case, unfortunately

mostly the case in CALCULUS ACCORDING TO THE REAL WORLD *texts* that $f\left(x_0 \oplus h\right)$ is a neighborhood of some output number, be it $f\left(x_0\right)$ or some other output number y_0 so that one can thicken the output level line into an **output level band**

CAUTION 4.1 One should absolutely *never* use the words "nearby outputs" as a short for outputs for nearby inputs because the output numbers $f\left(x_0 \oplus h\right)$ returned by the function f for $x_0 \oplus h$, that is for the input numbers in a neighborhood of x_0, need *not* make up a neighborhood of *any* output number y_0, let alone make up a neighborhood of the output number $f\left(x_0\right)$

Not even in the privacy of the reader's mind!

.

EXAMPLE 4.2. In EXAMPLE 1.11, even though the inputs 27.2 and 27.4 can be considered to be near, their outputs, respectively around $+70$ and -25, certainly cannot be considered anywhere near.

localize
local Input-Output rule
local I-O rule
local function
local Input-Output rule

4 Outputs *Near* A Given *Number*

Instead of getting **??** (**?? ??**, **??**), we will get Output *AT* A Given Number (Section 2, Page 199). But then, when the function is given by a global Input-Output rule, the issue becomes how do we get the local graph for inputs in a neighborhood of the given number?

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

As it happens, given a point, getting the output(s) for nearby inputs will be at the very heart of pretty much every investigation we will be doing.

Here, we will deal with the first step in the process which is to get the output(s), if any, returned by the function for inputs in a neighborhood of the given point.

A function f being given by a global I-O rule, the idea will be to **localize** the function f NEAR the given point, that is to get the ***local* Input-Output rule** (***local* I-O rule** for short) of a **local function**, that is of a (simpler) function that returns for inputs in a neighborhood of the given point the same output(s), if any, that the function f itself would return.

1. Output(s), if any, for inputs NEAR a given *number*. When the given point is a number x_0, the idea is to get the local I-O rule of a (simpler) local function f_{x_0} such that $f_{x_0}\Big(h\Big) = f(x_0 \oplus h)$, that is such that f_{x_0} will return for h the same output(s), if any, that f would return for $x_0 \oplus h$.

More precisely, ,

DEFINITION 4.2 A function f being given by a *global* Input-Output rule,

$$\underbrace{x}_{\text{Input}} \xrightarrow{\;f\;} \underbrace{\underbrace{f(x)}_{\text{Output}} = \textit{Global}\text{ expression in terms of } x}_{\text{Global input-output rule}}$$

the *local* Input-Output rule of the local function f_{x_0} is:

$$\underbrace{h}_{\text{Input}} \xrightarrow{f_{x_0}} \underbrace{\underbrace{f_{x_0}\big(h\big)}_{\text{Output}} = \textit{Local}\text{ expression in terms of } h}_{\text{Local input-output rule}}$$

Then,

Procedure 4.2 To get the output(s), if any, returned for inputs *near* a given *number* x_0 by a function f given by an global I-O rule $x \xrightarrow{f} f(x) = \text{Global expression in terms of } x$,

i. Declare that the inputs are *near* x_0 by using the local variable $x_0 \oplus h$ and writing the declaration $\Big|_{x \leftarrow x_0 \oplus h}$ to the right of each part of the global input-output rule:

$$x\Big|_{x \leftarrow x_0\oplus h} \xrightarrow{f} f(x)\Big|_{x \leftarrow x_0\oplus h} = \textit{Global}\text{ expression in terms of } x\Big|_{x \leftarrow x_0\oplus h}$$

ii. Execute the declaration, that is replace every occurence of the global variable x in the global input-output rule by the local variable $x_0 \oplus h$, to get the *individual* expression for $f(x_0 \oplus h)$:

$$x_0 \oplus h \xrightarrow{f} f(x_0 \oplus h) = \textit{Individual}\text{ expression in terms of } x_0 \oplus h$$

iii. Compute $\textit{Individual}\text{ expression in terms of } x_0 \oplus h$ to get:

$$\textit{Local}\text{ expression in terms of } h$$

iv. The *local* Input-Output rule for f *near* x_0 then is:

$$h \xrightarrow{f_{x_0}} \underbrace{f_{x_0}\big(h\big) = \textit{Local}\text{ expression in terms of } h}_{\textit{Local}\text{ input-output rule } \textit{near } x_0}$$

v. We then usually approximate $\textit{Local}\text{ expression in terms of } h$

Caution 4.2 *Local* expression in terms of h is different from *Global* expression in terms of x because, as the subscript x_0 is intended to indicate, the number x_0 has been "computed into" *Local* expression in terms of h.

Demo 4.2a To get the output(s), if any, returned for inputs NEAR -5 by the function $\mathcal{JILL}$ given by the global input-output rule $x \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x) = \dfrac{(-4\odot x)\oplus+7}{+2\odot(x\oplus+7)}$

i. We declare that the inputs are *near* -5 by using the local variable $-5\oplus h$ and writing the declaration $\Big|_{x\leftarrow -5\oplus h}$ to the right ot each part of the global input-output rule:

$$x\Big|_{x\leftarrow -5\oplus h} \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x)\Big|_{x\leftarrow -5\oplus h} = \frac{(-4\odot x)\oplus+7}{+2\odot(x\oplus+7)}\Bigg|_{x\leftarrow -5\oplus h}$$

ii. We execute the declaration, that is we replace every occurence of the global variable x in the global input-output rule by the local variable $-5\oplus h$ to get the *individual* expression for $\mathcal{JILL}\left(-5\oplus h\right)$

$$-5\oplus h \xrightarrow{\mathcal{JILL}} \mathcal{JILL}\left(-5\oplus h\right) = \underbrace{\frac{\left(-4\odot -5\oplus h\right)\oplus+7}{+2\odot\left(-5\oplus h\oplus+7\right)}}_{\text{Individual expression NEAR } -5}$$

iii. We compute $\dfrac{\left(-4\odot -5\oplus h\right)\oplus+7}{+2\odot\left(-5\oplus h\oplus+7\right)}$ that is we perform all the operations to get the local expression NEAR -5:

$$= \frac{\left(-4\odot -5\right)\oplus\left(-4\odot h\right)\oplus+7}{\left(+2\odot -5\right)\oplus\left(+2\odot h\right)\oplus(+2\odot+7)}$$

$$= \frac{+20 \oplus -4\,h \oplus +7}{-10 \oplus +2\,h \oplus +14}$$

$$= \underbrace{\frac{+27 \oplus -4\,h}{+4 \oplus +2\,h}}_{\text{Local expression NEAR } -5}$$

iv. The *local* Input-Output rule for $\mathcal{JILL}$ *near* -5 then is:

$$h \xrightarrow{\mathcal{JILL}_{-5}} \underbrace{\mathcal{JILL}_{-5}\big(h\big) = \frac{+27 \oplus -4\,h}{+4 \oplus +2\,h}}_{\text{Local input-output rule NEAR } -5}$$

v. We approximate $\frac{+27 \oplus -4\,h}{+4 \oplus +2\,h}$

$$= \frac{+27 \oplus \textit{small}}{+4 \oplus \textit{small}}$$

$$= +6.75 \oplus \textit{small}\ .$$

So, for inputs *near* -5, $\mathcal{JILL}$ returns outputs *near* $+6.75$.

DEMO 4.2b To get the output(s), if any, returned for inputs *near* -7 by the function $\mathcal{JILL}$ given by the global input-output rule $x \xrightarrow{\mathcal{JILL}} \mathcal{JILL}(x) = \frac{(-4 \odot x) \oplus +7}{+2 \odot (x \oplus +7)}$

i. We declare that the inputs are *near* -7 by using the local variable $-7 \oplus h$ and writing the declaration $\big|_{x \leftarrow -7 \oplus h}$ to the right ot each part of the global input-output rule:

$$x\Big|_{x \leftarrow -7\oplus h} \xrightarrow{\mathcal{JILL}(} \mathcal{JILL}(x)\Big|_{x \leftarrow -7\oplus h} = \frac{(-4 \odot x) \oplus +7}{+2 \odot (x \oplus +7)}\Bigg|_{x \leftarrow -7\oplus h}$$

ii. We execute the declaration, that is we replace every occurence of the global variable x in the global input-output rule by the local variable $-7 \oplus h$ to get the individual expression for $\mathcal{JILL}\left(-7 \oplus h\right)$:

$$-7 \oplus h \xrightarrow{\mathcal{JILL}} \mathcal{JILL}\left(-7 \oplus h\right) = \underbrace{\frac{\left(-4 \odot -7 \oplus h\right) \oplus +7}{+2 \odot \left(-7 \oplus h \oplus +7\right)}}_{\text{Individual expression NEAR } -7}$$

iii. We compute $\dfrac{\left(-4 \odot -7 \oplus h\right) \oplus +7}{+2 \odot \left(-7 \oplus h \oplus +7\right)}$ that is we perform all the operations to get the local expression NEAR -7:

$$= \frac{\left(-4 \odot -7\right) \oplus \left(-4 \odot h\right) \oplus +7}{\left(+2 \odot -7\right) \oplus \left(+2 \odot h\right) \oplus (+2 \odot +7)}$$

$$= \frac{+28 \oplus -4\,h \oplus +7}{-14 \oplus +2\,h \oplus +14}$$

$$= \underbrace{\frac{+35 \oplus -4\,h}{+2\,h}}_{\text{Local expression NEAR } -7}$$

iv. The *local* Input-Output rule for $\mathcal{JILL}$ *near* -7 then is:

$$h \xrightarrow{\mathcal{JILL}_{-7}} \underbrace{\mathcal{JILL}_{-7}\left(h\right) = \frac{+35 \oplus -4\,h}{+2\,h}}_{\text{Local input-output rule NEAR } -7}$$

v. We approximate $\dfrac{+35 \oplus -4\,h}{+2\,h}$

$$= \frac{+35 \oplus small}{small}$$

$$= large$$

So, for inputs *near* -7, $\mathcal{JILL}$ returns outputs *near* ∞.

2. Output(s), if any, for inputs NEAR ∞. When the given point is ∞, the idea is to get a (simpler) local function f_{∞} that will return for L the same output(s), if any, that f would return for L

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

PROCEDURE 4.3 To get the output returned *near* a given *point* by a function f given by an global I-O rule $x \xrightarrow{f} f(x) =$ Global expression in terms of x,

i. Declare that the output is to be *near* the given x_0 by writing to the right of the global input-output rule the declaration $\Big|_{x \leftarrow x_0}$:

$$x \xrightarrow{\quad f \quad} f(x) = \text{Global expression in terms of } x \Big|_{x \leftarrow x_0}$$

ii. Execute the declaration by replacing every occurence of x in the global input-output rule by the given input x_0 to get the individual expression for x_0:

$$x_0 \xrightarrow{\quad f \quad} f(x_0) = \text{Individual expression in terms of } x_0$$

iii. Compute the individual expression in terms of x_0, that is perform the operations in the individual expression to get:

$$\text{Individual expression in terms of } x_0 = y_0$$

iv. Format the input-output pair according to **?? ??** - **??** (**??**)

5 Local Input-Output Rule

In order to get the Start (Section 12, Page 193) NEAR a given point, we will need

DEFINITION 4.3 A function f being given by a global Input-Output rule,

$$\underbrace{x}_{\text{Input}} \xrightarrow{\;f\;} \underbrace{\underbrace{f(x)}_{\text{Output}} = \text{global expression in terms of } x}_{\text{Global input-output rule}}$$

the ***local*** **Input-Output rule**

- Near ∞

$$\underbrace{x}_{\text{Input}} \xrightarrow{\;f\;} \underbrace{\underbrace{f(x)}_{\text{Output}} = \text{global expression in terms of } x}_{\text{Global input-output rule}}$$

- Near a given *number* x_0

$$\underbrace{x}_{\text{Input}} \xrightarrow{\;f_{x_0}\;} \underbrace{\underbrace{f(x)}_{\text{Output}} = \text{global expression in terms of } x}_{\text{Global input-output rule}}$$

OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR

We already discussed in Expressions And Values (Section 4, Page 12) why, in the real world, we cannot use isolated numbers and in Neighborhoods - Local Expressions (Section 9, Page 44) that we need neighborhoods.

Local input-Output rule

In Start (Section 12, Page 193), we saw how to get global graphs from local graphs NEAR control points/

Here, we will see that to get the local graphs we need from ***Local* input-Output rules** to get outputs *near* a given point.

from which we will get local graphs which we will interpolate to get global graphs.

make a diagram here.

alluded to the heart of the matter in Neighborhoods - Local Expressions (Section 9, Page 44)

As hinted at in Start (Section 12, Page 193), the way we will operate is by interpolation of local graph graphs.

The question then is how to get the local graph NEAR a given point for the global I-O rule, that is how to comput outputs NEAR given numbers.

with computing outputs AT given numbers is that:

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**
OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

A major part of our work with functions given by input-output rules will be getting local graphs in order to:

- See Functions Given Graphically (Chapter 2, Page 93)
- Construct the global graph of the function to see The Looks Of Functions (Chapter 3, Page 141)

The first step towards getting local graphs for functions given by input-output rules will be to compute the output NEAR a given point.

The fact that global input-output rules involve a global expression in terms of a *number* will *not* prevent us from investigating a function NEAR a given point, be it ∞, 0, or x_0 because,

- near ∞, *RBC* will employ large-size numbers and therefore the large variable L
- near 0 *RBC* will employ small-size numbers and therefore the small variable h

local input-output pair
local input-output rule
local arrow pair

- near x_0 *RBC* will employ nearby mid-size numbers and therefore the NEAR mid-size number variable $x_0 \oplus h$

DEFINITION 4.4 Using the symbol V to stand for the appropriate one of the nearby variables for the given point: large variable L, small variable h, circa variable $x_0 \oplus h$, we have:

- For *graphing*, use the **local input-output pair**
$$\Big(V, \text{executed expression in terms of } V\Big)$$
- For *computing*, use the **local input-output rule**
$$\underbrace{f\Big(V\Big) = \text{executed expression in terms of } V}_{\text{Local input-output rule NEAR given point}}$$
- For *seeing*, use the **local arrow pair**
$$V \xrightarrow{f} \text{executed expression in terms of } V$$
- For *thinking*, use
$$V \xrightarrow{f} \underbrace{f\Big(V\Big) = \text{executed expression in terms of } V}_{\text{Local input-output rule NEAR given point}}$$

1. Near ∞

PROCEDURE 4.4 To get the outputs returned *near* ∞ by a function f given by an I-O rule $x \xrightarrow{f} f(x) =$ global expression in terms of x,

i. Declare that the input is a large-size indeterminate number by using the large variable L and writing the declaration $\Big|_{x \leftarrow L}$ to the right of the global input-output rule:
$$x \xrightarrow{f} f(x) = \text{global expression in terms of } x \Big|_{x \leftarrow L}$$

ii. Replace every occurence of x in the global input-output rule by the large variable L to get the **local input-output rule** *near* ∞:

$$L \xrightarrow{f} f\left(L\right) = \text{global expression in terms of } L$$

iii. Execute the global expression in terms of the relevant variable according to the rules in **?? ??** - **??** (**??**), that is do the operations in the global expression to get the **executed expression**

iv. Format according to DEFINITION 4.4 - Local formats (Page 212)

local executed expression
local input-output rule
local input-output pair
local input-output arrow pair

DEMO 4.3 To get the outputs returned for inputs *near* ∞ by the function $\mathcal{ZENA}$ given by the global input-output rule $x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \dfrac{x^{+2} \ominus +1}{x \oplus +3}$

i. We *declare* that the input is a large-size indeterminate number by writing the declaration $\Big|_{x \leftarrow L}$ to the right of the global input-output rule:

$$x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \frac{x^{+2} \ominus +1}{x \oplus +3}\Bigg|_{x \leftarrow L}$$

ii. We *replace* every occurence of x in the global expression by L to get the individual expression for L:

$$L \xrightarrow{\quad\mathcal{ZENA}\quad} \mathcal{ZENA}(L) = \frac{L^{+2} \ominus +1}{L \oplus +3}$$

iii. We *execute* the individual expression for L:

$$= \frac{L^{+2} \ominus +1}{L \oplus +3}$$

$$= \frac{L^{+2} \oplus [...]}{L \oplus [...]}$$

$$= L \oplus [...]$$

The last expression above *is* the executed expression.

iv. We *format* according to DEFINITION 4.4 - Local formats (Page 212)

- local Input-output pair $\Big(L, L \oplus [...]\Big)$

executed expression
local input-output rule
local input-output pair

- local input-output rule $\mathcal{ZENA}\left(L\right) = L \oplus [...]$
- local arrow pair $L \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\left(L\right)$
- local input-output rule $L \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\left(L\right) = L \oplus [...]$

2. Near 0

PROCEDURE 4.5 To get the outputs returned *near* 0 by a function f given by an I-O rule $x \xrightarrow{f} f(x) =$ global expression in terms of x,

i. Declare that the input is a small-size indeterminate number by using the small variable h and writing the declaration $\Big|_{x \leftarrow h}$ to the right of the global input-output rule:

$$x \xrightarrow{f} f(x) = \text{global expression in terms of } x\Big|_{x \leftarrow h}$$

ii. Replace every occurence of x in the global input-output rule by the small variable h to get the **local input-output rule** *near* 0:

$$h \xrightarrow{f} f\left(h\right) = \text{global expression in terms of } h$$

iii. Execute the global expression in terms of the relevant variable according to the rules in **?? ??** - **??** (**??**), that is do the operations in the global expression to get the **executed expression**

iv. Format according to DEFINITION 4.4 - Local formats (Page 212)

- For *graphing*, use the input-output pair
$$\left(h, \text{executed expression in terms of } h\right)$$
- For *computing*, use the equality
$$\underbrace{f\left(h\right) = \text{executed expression in terms of } h}_{\text{Local input-output rule NEAR } 0}$$
- For *seeing*, use the arrow pair
$$h \xrightarrow{f} \text{executed expression in terms of } h$$
- For *thinking*, use the **local input-output rule**

$$\underbrace{h \xrightarrow{f} f\big(h\big) = \text{executed expression in terms of } h}_{\text{Local input-output rule NEAR } 0}$$

DEMO 4.4 To get the outputs returned for inputs *near* 0 by the function $\mathcal{ZENA}$ given by the global input-output rule $x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \dfrac{x^{+2} \ominus +1}{x \oplus +3}$

i. We *declare* that the input is a small-size indeterminate number by using the small variable h and writing the declaration $\Big|_{x \leftarrow h}$ to the right of the global input-output rule:

$$x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \frac{x^{+2} \ominus +1}{x \oplus +3}\Bigg|_{x \leftarrow h}$$

ii. We *replace* every occurence of x in the global expression by h to get the individual expression for h:

$$h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(h) = \frac{h^{+2} \ominus +1}{h \oplus +3}$$

iii. We *execute* the individual expression for h:

$$= \frac{-1 \oplus h^2}{+3 \oplus +h}$$

$$= -\tfrac{1}{3} \oplus +\tfrac{1}{3^{+2}}h \oplus +\tfrac{8}{3^{+3}}h^{+2}$$

The last expression above *is* the executed expression.

iv. We *format* according to DEFINITION 4.4 - Local formats (Page 212)

$$\left(h, -\tfrac{1}{3} \oplus +\tfrac{1}{3^{+2}}h \oplus +\tfrac{8}{3^{+3}}h^{+2}\right)$$

$$\mathcal{ZENA}\big(h\big) = -\tfrac{1}{3} \oplus +\tfrac{1}{3^{+2}}h \oplus +\tfrac{8}{3^{+3}}h^{+2}$$

$$h \xrightarrow{\mathcal{ZENA}} = -\tfrac{1}{3} \oplus +\tfrac{1}{3^{+2}}h \oplus +\tfrac{8}{3^{+3}}h^{+2}$$

$$h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\big(h\big) = -\tfrac{1}{3} \oplus +\tfrac{1}{3^{+2}}h \oplus +\tfrac{8}{3^{+3}}h^{+2}$$

3. Near x_0

DEMO 4.5a To get the outputs returned for inputs *near* $+5$ by the function $\mathcal{ZENA}$ given by the global input-output rule $x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \dfrac{x^{+2} \ominus +1}{x \oplus +3}$

i. We *declare* that the input is an indeterminate number *near* $+5$ by writing the declaration $\Big|_{x \leftarrow +5 \oplus h}$ to the right of the global input-output rule:

$$x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \frac{x^{+2} \ominus +1}{x \oplus +3}\Bigg|_{x \leftarrow +5 \oplus h}$$

ii. We *replace* every occurence of x in the global expression by $+5 \oplus h$ to get the individual expression for $+5 \oplus h$:

$$+5 \oplus h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(+5 \oplus h) = \frac{+5 \oplus h^{+2} \ominus +1}{+5 \oplus h \oplus +3}$$

iii. We *execute* the individual expression for $+5 \oplus h$:

$$= \frac{+25 \oplus +10h \oplus +h^2 \ominus +1}{+5 \oplus +h \oplus +3}$$

$$= \frac{+24 \oplus +10h \oplus +h^2}{+8 \oplus +h}$$

$$= +3 \oplus +\tfrac{7}{8}h \oplus +\tfrac{1}{64}h^{+2} \oplus [...]$$

The last expression above *is* the executed expression.

Ok, so, why stop the dvision here? You will see in ?? ?? - ?? (??)

iv. We *format* the input-output pair:

- $\left(+5 \oplus h, +3 \oplus +\tfrac{7}{8}h \oplus +\tfrac{1}{64}h^{+2} \oplus [...]\right)$
- $\mathcal{ZENA}\left(+5 \oplus h\right) = +3 \oplus +\tfrac{7}{8}h \oplus +\tfrac{1}{64}h^{+2} \oplus [...]$
- $+5 \oplus h \xrightarrow{\mathcal{ZENA}} +3 \oplus +\tfrac{7}{8}h \oplus +\tfrac{1}{64}h^{+2} \oplus [...]$
- $+5 \oplus h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\left(+5 \oplus h\right) = +3 \oplus +\tfrac{7}{8}h \oplus +\tfrac{1}{64}h^{+2} \oplus [...]$

Demo 4.5b To get the outputs returned for inputs *near* -3 by the function $\mathcal{ZENA}$ given by the global input-output rule $x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \dfrac{x^{+2} \ominus +1}{x \oplus +3}$

i. We declare that the input is an *undeterminate number near* -3 by writing the declaration $\Big|_{x \leftarrow -3 \oplus h}$ to the right of the global input-output rule

$$x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \frac{x^{+2} \ominus +1}{x \oplus +3}\Bigg|_{x \leftarrow -3 \oplus h}$$

ii. We execute the declaration by replacing every occurence of x in the input-ouput rule by $-3 \oplus h$ to get the global expression in terms of numbers *near* -3

$$-3 \oplus h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\Big(-3 \oplus h\Big) = \frac{-3 \oplus h^{+2} \ominus +1}{-3 \oplus h \oplus +3}$$

iii. We compute the global expression in terms of $-3 \oplus h$, that is we perform the operations in the individual expression:

$$= \frac{+9 \oplus -6h \oplus h^2 \ominus +1}{-3 \oplus +3 \oplus h}$$

$$= \frac{+8 \oplus -6h \oplus h^2}{h}$$

$$= +8h^{-1} \oplus -6 \oplus h$$

$$= +8h^{-1} \oplus [...]$$

The last expression above *is* the executed expression.

iv. We format the input-output pair, that is we write:

- $\Big(-3 \oplus h, +8h^{-1} \oplus -6 \oplus h\Big)$
- $\mathcal{ZENA}(-3 \oplus h) = +8h^{-1} \oplus -6 \oplus h$
- $-3 \oplus h \xrightarrow{\mathcal{ZENA}} +8h^{-1} \oplus -6 \oplus h$
- $-3 \oplus h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\Big(-3 \oplus h\Big) = +8h^{-1} \oplus -6 \oplus h$

DEMO 4.5c To get the outputs returned for inputs *near* $+1$ by the function $\mathcal{ZENA}$ given by the global input-output rule $x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \dfrac{x^{+2} \ominus +1}{x \oplus -3}$

i. We declare that the outputs are to be for *numbers near* $+3$ by writing the declaration $\Big|_{x \leftarrow +1 \oplus h}$ to the right of the global input-output rule:

$$x \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}(x) = \frac{x^{+2} \ominus +9}{x \oplus -3}\Bigg|_{x \leftarrow +1 \oplus h}$$

ii. We execute the declaration by replacing every occurence of x in the input-ouput rule by $+1 \oplus h$ to get the global expression in terms of numbers *near* $+1$

$$+1 \oplus h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\big(+1 \oplus h\big) = \frac{+1 \oplus h^{+2} \ominus +1}{+1 \oplus h \oplus -3}$$

iii. We compute the global expression in terms of $+1 \oplus h$, that is we perform the operations in the global expression:

$$= \frac{+1 \oplus +2h \oplus h^2 \ominus +1}{+3 \oplus -3 \oplus h}$$

$$= \frac{+2h \oplus h^2}{h}$$

$$= +2 \oplus h$$

The last expression above *is* the executed expression.

iv. We format the input-output pair, that is we write:

- $\big(+1 \oplus h, +2 \oplus h\big)$
- $\mathcal{ZENA}(+1 \oplus h) = +2 \oplus h$
- $+1 \oplus h \xrightarrow{\mathcal{ZENA}} +2 \oplus h$
- $+1 \oplus h \xrightarrow{\mathcal{ZENA}} \mathcal{ZENA}\big(+1 \oplus h\big) = +2 \oplus h$

OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR OKsoFAR

4. A Few Words of Caution Though. Starting with Part II - Calculatable Functions (Page 197) though, functions will cease to be given by a global graph and will be given instead by an I-O rule

When a function will be given by an I-O rule instead of a global graph, though, we will have to be very careful before we use **??** because

In Subsection 4.3 - Local frame (Page 130) we discussed how to get a local graph when the function is given by a curve. When the function is given by an I-O rule, though, we start out with no global graph, though, and getting a local graph is much more complicated and will require the knowledge of the global graphs of 'power functions'.

Since $x_0 \oplus h$ is a thickening of x_0, it is most tempting and natural to think of $f\left(x_0 \oplus h\right)$ as a thickening of $f\left(x_0\right)$ but, even though it is "often" the case, unfortunately

mostly the case in CALCULUS ACCORDING TO THE REAL WORLD *texts* that $f\left(x_0 \oplus h\right)$ is a neighborhood of some output number, be it $f\left(x_0\right)$ or some other output number y_0 so that one can thicken the output level-line into an **output level-band**

> **CAUTION 4.3** One should absolutely *never* use the words "neighboring outputs" as a short for outputs for neighboring inputs because the output numbers $f\left(x_0 \oplus h\right)$ returned by the function f for $x_0 \oplus h$, that is for the input numbers in a neighborhood of x_0, need *not* make up a neighborhood of *any* output number y_0, let alone make up a neighborhood of the output number $f\left(x_0\right)$

Not even in the privacy of the reader's mind!

.

control point

EXAMPLE 4.3. In EXAMPLE 1.11, even though the inputs 27.2 and 27.4 can be considered to be near, their outputs, respectively around $+70$ and -25, certainly cannot be considered anywhere near each other.

6 Towards Global Graphs.

There is no general way to deal with functions given by I-O rules and how *RBC* will deal with functions given by I-O rules will depend entirely on the kind of expression in terms of x that appears in the I-O rule. In particular, there is no general procedure for getting the global graph of functions given by I-O rules. So here we will only be able to say some general things.

1. Foward problems

2. Reverse problems. When a function f is given by an input-output rule

$$x \xrightarrow{f} f(x) = \text{global expression in terms of } x$$

the reverse problem for a given y_0

$$f(x) = y_0$$

means to solve the *equation*

$$\text{global expression in terms of } x = y_0$$

However, since the necessary AGEBRA depends entirely on the kind of global expression in terms of x that the input-output rule involves, and therefore on what type of function f is, we will only be able to deal with reverse problems as we go along and study each type of functions.

3. Global graph. Altogether, ∞ and poles will be the inputs that we will call the **control points** for that function.

Chapter 2 - Functions Given Graphically (Page 93) showed how we need local graphs to *see* local function behaviors, but with functions given by an input-output rule we will have to use PROCEDURE 4.4 - Get output *near* ∞ from f given by an global I-O rule (Page 212) and then graph the local input-output rule.

$$\underbrace{x}_{\text{Input}} \xrightarrow{\quad f \quad} \underbrace{\underbrace{f(x)}_{\text{Output}} \quad = \quad \text{global expression in terms of } x}_{\text{Global input-output rule}}$$

and so, a function being given by an I-O rule, we will proceed in the following three steps:

a. Locate the points NEAR which we will need a local graph, that is:

- The control points, that is
- There will also the poles, if any, that is the input numbers for which the output is ∞
-

As we saw, there will always be ∞ because it is one of the control points, ∞ and at the very least the poles if any, of the given function.

b. We will have to find the local frames in which the local graphs will be.

c. We will have to find the shape of the local graph.

The reason that there is no simple PROCEDURE for getting local graphs is that:

Step **a** is a reverse problem which will require solving equations that will depend on the global expression in the global I-O rule that gives the function under investigation.

Step **b** of course has already been dealt with with **??** however CAUTION 2.1 will complicate matters.

Step **c** will depend on being able to approximate the given function.

4. Need for Power Functions. ,

So we will need local graphs for two purposes:

i. Get the global graph

ii. Get the local behaviors

So our approach will be:

i. Get the local graphs we will need to get the essential global graph

ii. Get the local graphs we need to get the needed behaviors

because no number of input-output pairs can almost never get us even an idea of the graph.

OKsoFarOKsoFarOKsoFarOKsoFarOKsoFarOKsoFarOKsoFarOKsoFarOKsoFarOKsoFar

==============OK SO FAR ===============

]

Part III

Appendices

Appendix A

Dealing With Decimal Numbers

=======Begin HOLDING=======

1 Computing With Non-Zero Numbers

What makes the calculus language appropriate for computations is the use of **expressions**

Unfortunalely, defining expressions formally is actually complicated and certainly beyond the scope of *this* text.

Fortunately, as we will now see, the expressions that *we* will be using will be quite simple so that we can safely leave the formal definitions to MATHEMATICAL LOGIC. (https://en.wikipedia.org/wiki/Expression_(mathematics))

There are also several issues we need to bring up that all have to do with the fact that computing with *signed* numbers automatically involves computations with *plain* numbers, thereby creating a risk of confusion.

1. Comparing (non-zero) numbers. The most important matter to keep in mind is that: (Will go to Dealing With Decimal Numbers (Appendix A, Page 225))

⊕
⊖
addition
subtraction

i. Comparing *signed* numbers (**?? ??** - **??** (**??**)) is quite different from comparing *plain* numbers—even though we use the same symbols, $<$, $>$, and $=$, with both plain numbers and signed numbers:

- *Positive* numbers compare the *same* way as their sizes,
- *Negative* numbers compare the *opposite* way from their sizes,

and

- Non-zero numbers with opposite signs compareindependently of their sizes: *negative* numbers are smaller than *positive* numbers *regardless* of their sizes.

and

ii. The everyday use of *plain* numbers with *words* instead of *symbols* to indicate the orientations can make using the words *larger than*, *smaller than* and *equal to* quite confusing.

EXAMPLE A.1. In everyday language, we say that

A \$700 *expense* is *larger* than a \$300 *expense*

because 700 is larger than 300 but the word *expense* cannot be seen as just meaning — because

-700 is *smaller* than -300.

> **CAUTION A.1** Larger than, smaller than, equal to have different meanings depending on whether we are comparing *signed* numbers or comparing *plain* numbers.

.

2. Adding and subtracting (non-zero) numbers. Notice that we have been using + and − not only as symbols for *addition* and *subtraction* of *plain* numbers, both *whole* and *decimal*, in spite of these being already quite different sets of numbers, but now also as symbols to distinguish positive numbers from negative numbers.

So, to avoid confusion as much as possible,

?? ?? - ?? (??) uses the symbols ⊕ and ⊖.

> **DEFINITION A.1** **⊕ and** ⊖, read "oplus" and "ominus", will be the symbols we will use for addition and subtraction of *signed* numbers.

Which is presumably why, say, $+13.73$ and -78.02 used to be written as $^{+}13.73$ and $^{-}78.02$ since $^{+}13.73 -^{-} 78.02$ has the same advantages as $+13.73 \ominus -78.02$.

While the main reason for the ○ around the symbols + and − is to remind us to take care of the signs, another benefit is that using ⊕ and ⊖ lets us avoid having to use lots of parentheses.

$\odot$

$\_\_\_$

EXAMPLE A.2. Instead of writing the standard expressions

$-23.87 + (-3.03), \quad -44,29 - (+22.78), \quad +12.04 - (-41.38)$

we will write the expressions as:

$-23.87 \oplus -3.03, \quad -44,29 \ominus +22.78, \quad +12.04 \ominus -41.38$

which makes it clear without using parentheses which are symbols for calculations and which are symbols for signs.

THEOREM A.1 Opposite numbers add to 0:
Two numbers are opposite iff the two numbers add-up to 0.

3. Multiplying and dividing (non-zero) numbers.

i. While we could use the symbol $\otimes$ for the multiplication of *signed* numbers, we will use the symbol $\odot$ because the symbol $\cdot$ is the usual practice in CALCULUS texts.

ii. Similarly, while we could use the symbol $\oplus$ for the division of *signed* numbers, we will use the fraction bar $\_\_$ because it is the usual practice in CALCULUS texts.

?? ?? - ?? (??) uses the symbols $\odot$ and $—$.

For good reasons as you will see. And no circle around that one either!

EXAMPLE A.3.

$+2 \odot +5 = +10, \quad +2 \odot -5 = -10, \quad -2 \odot +5 = -10, \quad -2 \odot -5 = +10$

$\frac{+12}{+3} = +4, \quad \frac{+12}{-3} = -4, \quad \frac{-12}{+3} = -4, \quad \frac{-12}{-3} = +4,$

THEOREM A.2 Reciprocal *non-zero* numbers multiply to $+1$
Two numbers are reciprocal ifff the two numbers multiply to $+1.0$

4. Operating with more than two (non-zero) numbers With three numbers, let's call them Number One, Number Two, Number Three (which may or may not be the same) and two operations, let's call them operation one and operation two (which may or may not be the same):

Number One operation one Number Two operation two Number Three

the overall result will usually depend on the order in which we perform the operations.

EXAMPLE A.4. The two computations for the expression $-3 \ominus +5 \ominus -7$:

a. $\underbrace{\underbrace{-3 \ominus +5}_{-8} \ominus -7}_{-1}$ **b.** $\underbrace{-3 \ominus \underbrace{+5 \ominus -7}_{+12}}_{-15}$

rule

EXAMPLE A.5. The two computations for the expression $-3 \odot +5 \oplus -7$

a. $\underbrace{\underbrace{-3 \odot +5}_{-15} \oplus -7}_{-22}$ **b.** $\underbrace{-3 \odot \underbrace{+5 \oplus -7}_{-2}}_{+6}$

So, another reason to use ⊕, etc as that keeps the number of parentheses down.

i. So, to indicate which operation(s) is/are intended to be performed *ahead of the other(s)*, one uses parentheses, ( ).
However, when one attempts to *minimize* the number of parentheses, stating "rules" to indicate the order in which operations are to be performed is actually a surprisingly complicated issue. (See https://en.wikibooks.org/wiki/Basic_Algebra/Introduction_to_Basic_Algebra_Ideas/Order_of_Operations and/or https://en.wikipedia.org/wiki/Order_of_operations)

Because we will want to focus on the CALCULUS rather than on the ALGEBRA.

ii. So, in order to keep matters as simple as possible, this text will always use however many parentheses will be necessary and we will just agree that the only

In other words, no PEMDAS, no BEDMAS, no BODMAS, no BIDMAS. (https://en.wikipedia.org/wiki/Order_of_operations)
Just WYSIWYG.

> **AGREEMENT A.1** Computable expressions are expressions that, after parentheses surrounding a *single* number (if any) have been removed,
> **Rule A.** Do *not* include only *one* parenthesis (left or right),
> **Rule B.** May include two surrounding parentheses.

EXAMPLE A.6. In EXAMPLE 0.16, using AGREEMENT B.1 - 'Number' (without qualifier) (Page 249),

a. With $(-3 \ominus +5) \ominus -7$,

- We cannot perform $\ominus$ as the expression $+5) \ominus -7$ breaks **Rule A**.
- We can perform $\ominus$ as the expression $(-3 \ominus +5)$ complies with **Rule B**.

The computation would thus be writen:

$$(-3 \ominus +5) \ominus -7 \underbrace{= (-8) \ominus -7}_{\text{Step can be skipped}} = -8 \ominus -7 = -1$$

b. With $-3 \ominus (+5 \ominus -7)$,

- We cannot perform $\ominus$ as the expression $-3 \ominus (+5$ breaks **Rule A**.
- We can perform $\ominus$ as the expression $(+5 \ominus -7)$ complies with **Rule A** and **Rule B**. The computation would thus be written:

$$-3 \ominus (+5 \ominus -7) \underbrace{= -3 \ominus (+12)}_{\text{Step can be skipped}} = -3 \ominus +12 = -15$$

EXAMPLE A.7. In EXAMPLE 0.17 (Page 11) 0.17, using AGREEMENT B.1 - 'Number' (without qualifier) (Page 249),

picture ruler equidistant tickmark scale origin

a. With $(-3 \odot +5) \oplus -7$:

- We cannot perform $\oplus$ as the expression $+5) \oplus -7$ breaks **Rule A**.
- We can performe $\odot$ as the expression $(-3 \odot +5)$ complies with **Rule B**.

The computation would thus be writen:

$$(-3 \odot +5) \oplus -7 \underbrace{= (-15)}_{\text{Step can be skipped}} \oplus -7 = -15 \oplus -7 = -22$$

b. With $-3 \odot (+5 \oplus -7)$:

- We cannot perform $\odot$ as the expression $-3 \odot (+5$ breaks **Rule A**.
- We can perform $\oplus$ as the expression $(+5 \oplus -7)$ complies with **Rule B**.

The computation would thus be written:

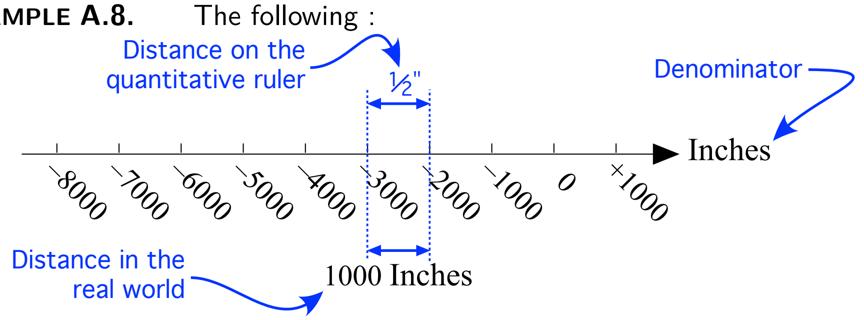
$$-3 \odot (+5 \oplus -7) \underbrace{= -3 \odot (-2)}_{\text{Step can be skipped}} = -3 \odot -2 = +6$$

2 Picturing Numbers

To **picture** numbers, RBC will employ **rulers** which, in the calculus language, are essentially just what goes by the name of "ruler" in ordinary English, that is an oriented straight line with **equidistant tickmarks** together with a denominator.

i. Scale. The **scale** of a ruler is, because tickmarks are equidistant, the ratio of *any* distance on the ruler to the corresponding distance in the real world (`https://en.wikipedia.org/wiki/Scale_(represent)`)

EXAMPLE A.8. The following :



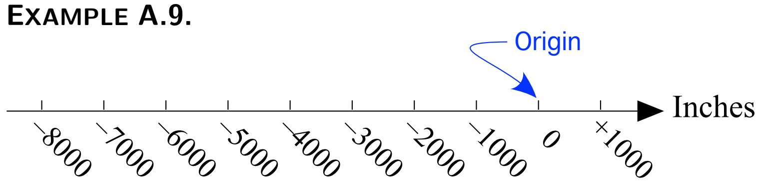
is a *quantitative* ruler whose scale is $\frac{\frac{1}{2}\text{inch}}{1\,000\text{ inch}} = \frac{1\text{ inch}}{2\,000\text{ inches}} = \frac{1\,\cancel{\text{inch}}}{2\,000\,\cancel{\text{inches}}} = \frac{1}{2\,000}$

ii. Origin. Rulers must have a tickmark labeled 0 as an **origin**,

0 for Origin as well as for zero.

sign
side
positive number
negative number
symmetrical

EXAMPLE A.9.

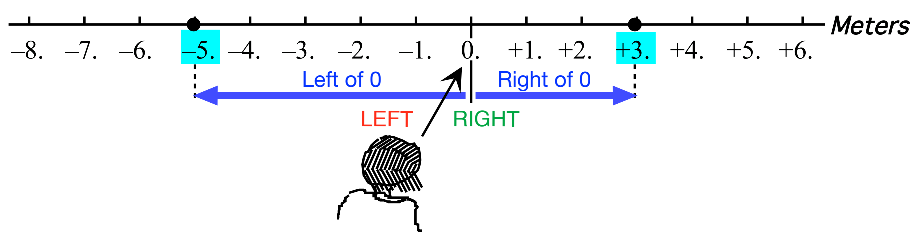


To know where the origin is is necessary because:

- The **sign** in a signed number says which **side** of the origin the signed number is—as seen when facing 0—and we will agree that

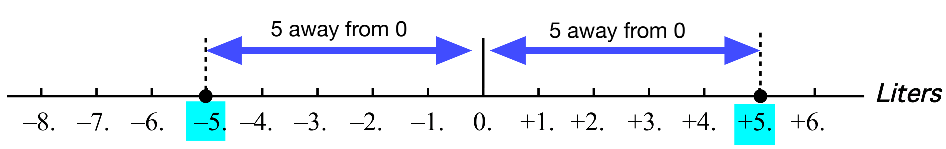
> **AGREEMENT A.2** When facing 0,
> ▸ **Positive numbers** (+ sign) will be to the *right* of the origin,
> ▸ **Negative numbers** (− sign) will be to the *left* of the origin 0.

EXAMPLE A.10. On a ruler,
Since Sign −5 is −, the number −5 is tickmarked *left* of 0.
Since Sign +3 is +, the number +3 is tickmarked *right* of 0.



- The size of a number says *how far* from 0 the number is on a ruler. Since opposite numbers have the same size, opposite numbers are **symmetrical** relative to the origin.

EXAMPLE A.11. The numbers −5.0 and +5.0 have the same *size*, namely 5.0, so they are equally far from 0:



3 Real World *Numbers* - Paper World *Numerals*

paper world
entity
numeral phrase
numerator
numeral
magnitude
quantitative information
denominator
essence
qualtative information
collection
item
whole number
count

With heavy reminders of to which world each word belongs!

Separating what is happening in the real world from what is happening in the **paper world** of a text is not easy so this section will use the terminology used in MODEL THEORY and LINGUISTICS. And since it is impossible to exhibit in the paper world the real world **entities** we will want to calculate about, we will use paper world *drawings* as *stand-ins* for real world entities:

There are two kinds of real world entities which we will both denote with paper world **numeral phrases** consisting of:

- A **numerator** using **numerals** (https://en.wikipedia.org/wiki/Numeral_(linguistics)) to provide the **magnitude** of the entity. (**Quantitative information.**)

and

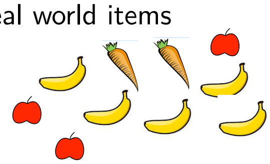
- A **denominator** using words to provide the **essence** of the entity. (**Qualtative information.**)

However, the two kinds of real world entities are different enough that we will have to use two different kinds of paper world *numerals* in the numerators.

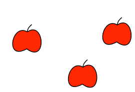
1. Magnitude of collections of items.

i. **Real world.** Since we get a real world **collection** of *identical* real world **items** just by gathering the real world items, determining how many real world items there are in a collection is simple: we get the **whole number** of real world items in the collection just by **counting** the real world items in the collection. .

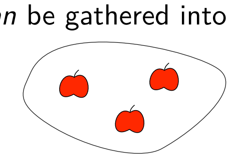
EXAMPLE A.12. The real world items



are *not* all the same and so *cannot* be gathered into a real world collection but the real world items

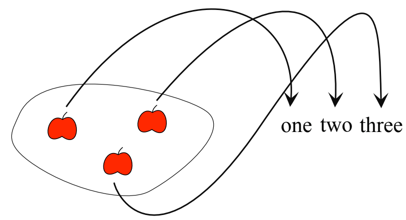


are all the same and so *can* be gathered into a real world collection:



plain whole numeral
unit
decimal number

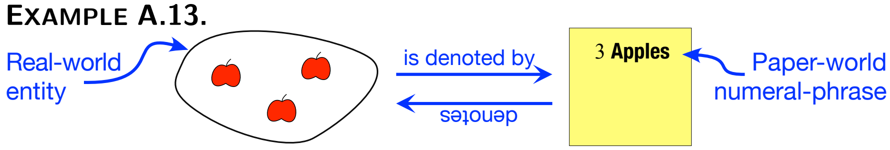
and we get the whole number by *counting* the items:



ii. **Paper world.** Collections of items are then denoted by paper world numeral phrases in which:

- The paper world numerator is the paper world **plain whole numeral** which says *how many* items there are in the collection, that is which denotes the real world whole number of items in the real world collection,
- The paper world denominator is the paper world word which says *what kind* of items are in the collection, that is which denotes the kind of real world items in the real world collection.

EXAMPLE A.13.



where:

· The *numeral* 3 says *how many* items in the collection,

and where

· The *word* **Apple** says *what kind* items in the collection.

2. Magnitude of amounts of stuff.

i. **Real world.** Since *stuff* comes *in bulk*, determining *how much* stuff there is in an amount of stuff is much more complicated than deciding *how many* items there are in a collection of items because, in order to determine how much stuff there is in a real world amount of stuff, we first need a real world **unit** of that stuff. Only then can we determine the **decimal number** of units in the amount of stuff.

Which is why "The Weights and Measures Division promotes uniformity in U.S. weights and measures laws, regulations, and standards to achieve equity between buyers and sellers in the marketplace." *(https://www.usa.gov/federal-agencies/weights-and-measures-division)*

EXAMPLE A.14. Milk is *stuff* we drink and before we can say *how much* milk we have or want, we must have a real world *unit* of milk, say *liter* of milk or *pint* of milk.

ii. **Paper world.** Amounts of stuff are then denoted by paper world numeral phrases in which:

- The paper world numerator is the paper world **plain decimal numeral** which says *how much* stuff there is in the amount of stuff, that is, more precisely, the plain decimal numeral in which the **decimal pointer** indicates which **digit** corresponds to the unit of stuff in the denominator, which denotes the real world decimal number of units of stuff in the amount of stuff.
- The paper world denominator is the paper world word which says *what kind* of stuff in the amount of stuff and what unit of stuff.

plain decimal numeral
decimal pointer
digit
orientation
signed whole numeral
signed decimal numeral

Which points to its left.

EXAMPLE A.14. (Continued) *Then* we may say we have or want, say, 6.4 *liters* of milk or, say, 3 pints of milk.

It should be noted that decimal numerals work hand in hand with the METRIC SYSTEM of units while US Customary units usually require *fractions*, $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, etc and mixed fractions.

Although panels on interstate roads have begun to show such things as 3.7 *Miles.*

3. Orientation of entities. Numerators can provide more information than just the magnitude of the entity, that is about the whole number of items or the decimal number of units of stuff, and can also provide information about the **orientation** of the entity by using **signed whole numerals** and **signed decimal numerals** instead of plain whole numerals and plain decimal numeral

4. Concluding remarks.

i. Since decimal numeral denote amounts of stuff while whole numerals denote collections of items, we absolutely need to distinguish *decimal* numerals from *whole* numerals.

EXAMPLE A.15. We need to distinguish the *decimal* numeral 27. which we would denote an amount of *stuff* from the *whole* numeral 27 which would denote a collection of *items*.

So, it would be tempting to agree that "The decimal point will *never* go without saying in this text." but, unfortunately, this is not really sustainable.

Told him it wouldn't! Didn't believe me! Wasted a lot of time trying anyway.

So, like everybody, we will have to agree that

> **AGREEMENT A.3** will often go without saying and we will often leave it to the reader to decide which kind of numeral is intended.

qualifier

Of course, sales people would write $11.99!

EXAMPLE A.16. When using money, pennies may or may not be beside the point:

- We are more likely to write $12.00 than $12

but

- We are more likely to write $7 000 000 than $7 000 000.00.

ii. Altogether then, since the kind of numeral used in the numerator depends on:

A. Whether the real world entity we want to denote is:

- A collection of items

or

- An amount of stuff

and also on:

B. Whether the **information** we want about the real world entity is:

- The magnitude of the entity *alone*,

or

- The magnitude *and* the orientation of the entity,

the word numeral should always be used with one of the following **qualifiers**

In fact, mathematicians, scientists, and engineers also use many other kinds of 'numbers' for many other kinds of entities. (https://en.wikipedia.org/wiki/Number)

	Collections	Amounts
Magnitude *only*	plain whole	plain decimal
Magnitude *and* orientation	signed whole	signed decimal

EXAMPLE A.17.

- 783 043 is a plain whole numeral which may denote a collection of *people*,
- 648.07 is a plain decimal numeral which may denote an amount of *money*,
- $-547\,048\,308$ and $+956\,481$ are signed whole numerals,
- $-137.048\,8$ and $+0.048\,178$ are signed decimal numerals.

And, since, as mentioned almost from the outset of ?? - Preface You *Don't* Have To Read (Page xi), this text assumes that the reader knows how to "compare, add/subtract, multiply/divide" signed decimal 'numbers', we will take the qualifiers plain/signed and whole/decimal to have been defined.

But you can always click on Appendix C - Localization (Page 251)

iii. However,

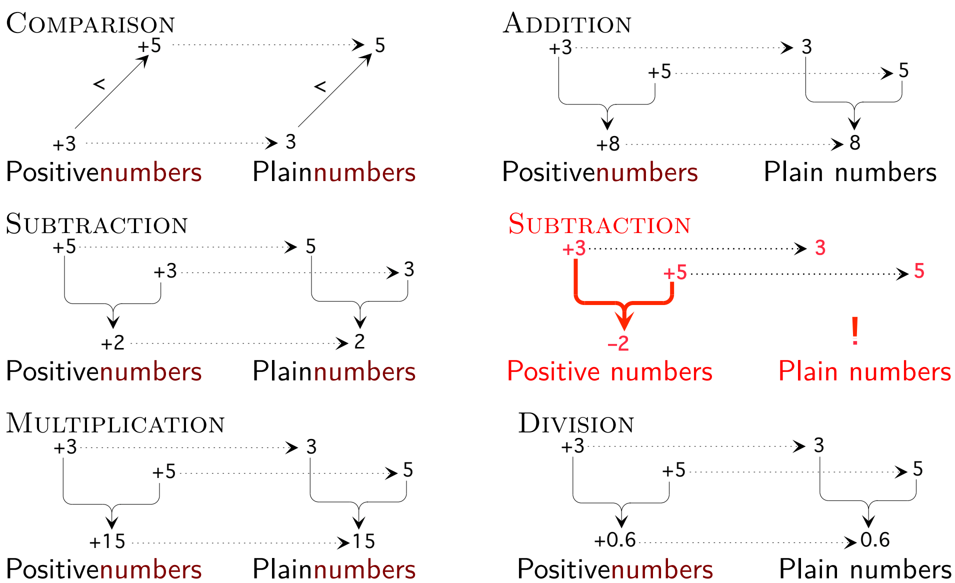
CAUTION A.2 While DISCRETE MATHEMATICS deals with collections of items, CALCULUS deals only with amounts of stuff and we will use whole numerals only occasionally and then mostly as an explanatory backdrop for decimal numbers.

4 Things To Keep In Mind

1. *Positive* numbers vs. *plain* numbers. Except for *subtraction*, computing with *positive* numbers goes exactly the same way as computing with the *plain* numbers that are their sizes..

And in only half the cases at that.

EXAMPLE A.18.



So it is tempting to skip the + sign in front of *positive* numbers as "going without saying". But then sentences lose their symmetry.

EXAMPLE A.19. The sentences

- "The opposite of $+5$ is -5" and "The opposite of -3 is $+3$"

are both nicely symmetric while the sentences

- "The opposite of 5 is -5" and "The opposite of -3 is 3"

both lack symmetry.

But then experience shows that skipping the + sign in front of *positive* numbers can lead to *ignoring* the difference between positive numbers and plain numbers and *that* leads to misunderstanding and mistakes because

- while working with *plain* numbers we can just focus on the numbers we are working with, *In fact, negative numbers were called* **absurd** *numbers for a long time until* "Calculus made negative numbers necessary."*(https://en.wikipedia.org/wiki/Negative_number#History)*

absurd
collection
item
count
plain
whole
natural
positive integer

In other words, you get exactly what you see, no more, no less.

- when working with *positive* numbers we have to keep constantly in mind that the numbers we are working with have a sign, namely +, and therefore have opposites, namely negative numbers.

And so, in order to help distinguishing signed numbers from plain numbers and more individually positive numbers from their sizes, in this text:

In other words, in this text, no sign does NOT *mean* positive *but* plain *and therefore NO opposite.*

AGREEMENT A.4 will *never* go without saying.

EXAMPLE A.20. We will always distinguish, for instance,

- The *positive* number $+51.73$ from the *plain* number 51.73 which is the *size* of $+51.73$. (As well as the *size* of -51.73)
- The *positive* number $+64\,300$ from the *plain* number $64\,300$ which is the *size* of $+64\,300$. (As well as the *size* of $-64\,300$)

2. Symbols vs. words. Another issue is that, in everyday language, instead of using signed numbers we still tend to use plain numbers with *everyday words* instead of symbols to denote the orientation.

Yet, even banks, *which used to use plain numbers in two columns, one for debits, one for credits, now use signed numbers in a single column.*

EXAMPLE A.21. We often use *words* like credit and debit, left and right, up and down, income and expense, gain and loss, incoming and outgoing, etc instead of the *symbols* + and − to denote the *orientation* and using *plain* numbers to denote the *size*.

5 Plain Whole Numbers

Remember that words between single quotes *will be explained when their time comes.*

Because we can deal with **collection** of **items** *one by one*, describing how many items there are in a collection is easy: just **count** the items in the collection. Then, how many items there are in the collection will be given by a **plain** (as opposed to 'signed') **whole** (as opposed to 'decimal') number.

.

EXAMPLE A.22. Apples are *items*. (We can eat apples one by one.) To say how many  are in the collection  we *count* them that is we point successively at each  while singsonging "one, two, three".

But not in this *text.*

how many
how much
decimal
amount
stuff
orientation
magnitude
signed
size
quantitative
absolute value
numerical value
modulus
norm
| |

> **LANGUAGE A.1** **Plain whole** numbers are also called **counting** numbers or **natural** numbers (https://en.wikipedia.org/wiki/Natural_number)—and, *incorrectly*, 'positive integers'.

===================================

decimal (as opposed to whole

An **amount** of **stuff** we can deal with only *in bulk*

orientation

magnitude that is how many items in the collection or how much stuff in the amount

> **LANGUAGE A.2**The word orientation is *not* too good but the words "*direction*" and "way" aren't either.

===================================

A lot of times, describing *how many* items we have or want in a collection or *how much* stuff we have or want in an amount of stuff is not enough and we also need to describe the *orientation* of the collection of items or of the amount of stuff: up/down, left/right, in/out, etc.

EXAMPLE A.23. How many people are *going into* or *coming out* of a building usually depends on the time of the day.
At least for the rest of us, how much money is *coming into* or *going out* of our bank account usually depends on the day of the month.

1. Size and sign. So, both **signed** (as opposed to plain) *whole* numbers and signed (as opposed to plain) *decimal* numbers carry *two* kinds of information:

- The **size** of a signed number (whole or decimal) is the **quantitative** information which is given by the plain whole number that describes *how many* items there are in the collection or the plain decimal number that describes *how much* stuff there is in the amount.

  > **LANGUAGE A.3** **Size** is called **absolute value** in most textbooks but some use **numerical value** or **modulus** or **norm**.

  The standard symbol for size is $|\ \ |$ but we will not use it and just write size of.

sign
qualitative
+
−
positive
negative
integers
the same
the opposite
opp

EXAMPLE A.24. Instead of $|-3| = 3$ we will write: size $-3 = 3$.

- The **sign** of a signed-number (whole or decimal) is the **qualitative** information which is given by + or −, the symbols that describe the *orientation* of the collection or of the amount, up/down, left/right, in/out, after a decision has been made as to which orientation is to be symbolized by + and therefore which by −. Then,
  Positive (whole or decimal) numbers are the signed numbers whose sign is +,
  Negative (whole or decimal) numbers are the signed numbers whose sign is −.

EXAMPLE A.25. $+17.43$ **Dollars** specifies a real world transaction:
- The *size* of $+17.43$, 17.43, describes the *magnitude* of the transaction,
- The *sign* of $+17.43$, $+$, describes the *orientation* of the transaction.

LANGUAGE A.4 **Signed *whole*** numbers are usually called **integers**.

But how could a plain whole number ever be called a positive integer?

Two signed numbers are:
- **the same** whenever they have the *same* size *and the same* signs. (So, when one is positive, the other has to be positive and vice versa.)
- **the opposite** whenever they have the *same* size *but different* signs. (So, when one is positive, the other has to be negative and vice versa.)

We will use **opp** as shorthand for opposite of.

EXAMPLE A.26.
opp $(+32.048) = (-32.048)$ opp $(-32.048) = (+32.048)$

=======End LOOK UP =======

As implied by the title, operating on *plain* numbers, whole and decimal, is assumed to be known and this Appendix deals only with the complications brought about by the signs.

.

- ?? ?? - ?? (??) • ?? ?? - ?? (??) • ?? ?? - ?? (??)

6 Comparing.

The symbols, $<$, $>$, $=$, $\leqq$, $\geqq$, are used for *both* **(plain) comparisons** *and* **(signed) comparisons**

$<$
$>$
$=$
$\leqq$
$\geqq$
comparison (plain)
comparison (signed)
larger-than (plain)
smaller-than (plain)
equal-to (plain)
not-equal-to (plain)
larger-than-or-equal-to (plain)
smaller-than-or-equal-to (plain)
larger-than (signed)
smaller-than (signed)
equal-to (signed)
not-equal-to (signed)
larger-than-or-equal-to (signed)
smaller-than-or-equal-to (signed)
smaller than
larger than

DEFINITION A.2 Given the signed numbers x_1 and x_2,

- When x_1 and x_2 are *both positive*,

$$x_1 > x_2 \text{ iff Size } x_1 > \text{Size } x_2$$
$$x_1 < x_2 \text{ iff Size } x_1 < \text{Size } x_2$$
$$x_1 = x_2 \text{ iff Size } x_1 = \text{Size } x_2$$

- When x_1 and x_2 are *both negative*,

$$x_1 > x_2 \text{ iff Size } x_1 < \text{Size } x_2$$
$$x_1 < x_2 \text{ iff Size } x_1 > \text{Size } x_2$$
$$x_1 = x_2 \text{ iff Size } x_1 = \text{Size } x_2$$

- When x_1 and x_2 have opposite signs,

$x_1 < x_2$ iff x_1 is negative (and therefore x_2 is positive)
$x_1 > x_2$ iff x_1 is positive (and therefore x_2 is negative)

larger-than

smaller-than

equal-to

not-equal-to

larger-than-or-equal-to

smaller-than-or-equal-to

larger-than

smaller-than

equal-to

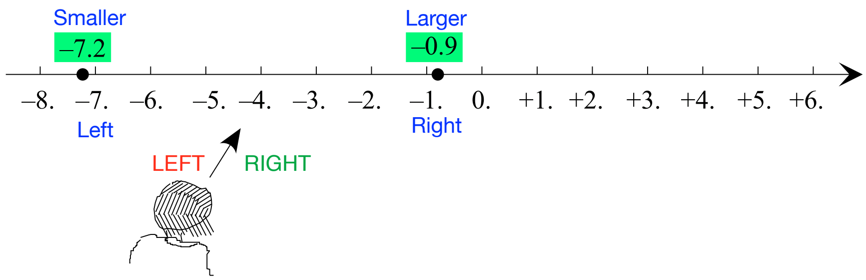
not-equal-to

larger-than-or-equal-to

smaller-than-or-equal-to

The easiest way is to picture the two numbers on a quantitative ruler and then, because of **?? ??** - **??** (**??**) , the number to *our left* will be **smaller than** the number to *our right* and the number to *our right* will be **larger than** the number to *our left*.

EXAMPLE A.27. Given the numbers -7.2 and $-0.9.$ we have add

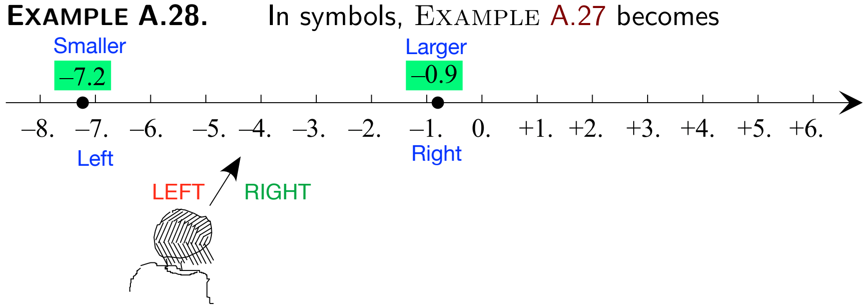


so –7.2 is smaller than –0.9 and –0.9 is larger than –7.2

The *standard* symbols for sign-size-comparisons of *all four kinds* of numbers are:

Sign-size-comparisons	Symbols
equal to	$=$
not equal to	$\neq$
smaller than	$<$
smaller than or equal to	$\leqq$
larger than	$>$
larger than or equal to	$\geqq$

Example A.28. In symbols, Example A.27 becomes



so $-7.2 < -0.9$ as well as $-0.9 > -7.2$

7 Adding and Subtracting

. To **add**

In this text, for reasons explained in **?? ??** - **??** (**??**), when dealing with signed numbers, we will use the word oplus instead of the word add which we will reserve for plain numbers.

we will use the symbol $\oplus$

addition
To **subtract** a number we oplus its opposite instead.
subtraction

subtract
multiply
divide
reciprocal (plain)

8 Multiplying and Dividing

. To **multiply**

MEMORY A.1 Multiplication and Division of Signs

	+	−
+	+	−
−	−	+

To **divide**

1. Reciprocal of a number.

i. The **reciprocal** of a *plain* number is 1. divided by that number. (https://www.mathsisfun.com/reciprocal.html). So:

i. Reciprocal 1. = 1.

ii. The reciprocal of 1 followed or preceded by 0s is easy to get: read the number you want the reciprocal of and insert/remove "th" accordingly,

EXAMPLE A.29.

Reciprocal 1 000. = 1 thousand th = 0.001

Reciprocal 0.000 001 = 1 million ~~th~~ = 1 000 000.

iii. The reciprocal of other numbers needs to be calculated and, for most, we may as well use a calculator.

EXAMPLE A.30.

$$\text{Reciprocal } 4.00 = \frac{1.00}{4.00} = +0.25$$ (Hopefully by hand.)

$$\text{Reciprocal } 0.89 = \frac{1.00}{0.89} = 1.13$$ (Use a calculator.)

$$\text{Reciprocal } 2.374 = \frac{1.00}{2.374} = 0.421$$ (Use a calculator.)

reciprocal (signed)

An important property of reciprocals is that:

MEMORY A.2 Sizes of *plain* reciprocal numbers
The larger a *plain* number is, the smaller its reciprocal will be,
The smaller a *plain* number is, the larger its reciprocal will be.

Proof. □

EXAMPLE A.31.

ii. The **reciprocal** of a *signed* number is +1. divided by that number. So, getting the reciprocal of a signed number involves Memory A.1 - Multiplication and Division of Signs (Page 241) which complicates matters:

EXAMPLE A.32.

$$\text{Reciprocal } +1\,000. = +1 \text{ thousand th} = +0.001$$

$$\text{Reciprocal } -0.000\,001 = -1 \text{ millionth} = -1\,000\,000.$$

$$\text{Reciprocal } +4.00 = \frac{+1.00}{+4.00} = +0.25 \text{ (Hopefully by hand.)}$$

$$\text{Reciprocal } -0.89 = \frac{+1.00}{-0.89} = -1.13 \text{ (Use a calculator.)}$$

$$\text{Reciprocal } -2.374 = \frac{+1.00}{-2.374} = -0.421 \text{ (Use a calculator.)}$$

In particular, even just *stating* the extension of Memory A.2 - Sizes of *plain* reciprocal numbers (Page 242) to signed numbers is a bit complicated and is much easier done in Subsection 12.1 - substart (Page 193).

To be specific: **?? ??** - **??** *(??).*

real number

Appendix B

Real Numbers

The sole purpose of this Appendix is to explain why this text is using signed decimal numbers instead of the so-called **real numbers** to be found in most CALCULUS texts, and what using in this text real numbers instead of signed decimal numbers would have entailed.

1 What *are* the real numbers?

1. Title. Even though most college mathematics textbooks claim to *use* real numbers, the closest they ever come to *defining* real numbers is something along the lines of "*a real number is a value of a continuous quantity that can represent a distance along a line.*" (https://en.wikipedia.org/wiki/Real_number or https://math.vanderbilt.edu/schectex/courses/thereals/)

Which, one has to admit, isn't particularly enlightening. Moreover, the wording in Wikipedia keeps changing with time! A sign of unease?

And of course, there is a very good reason for this vagueness (https://en.wikipedia.org/wiki/Vagueness_and_Degrees_of_Truth): in contrast with signed decimal numbers, real numbers are so *extremely* complicated to define that it is only done in REAL ANALYSIS.

> "*The real number system* $(\mathbb{R}; +; \cdot; <)$ *can be defined axiomatically* [...] *There are also many ways to construct "the" real number system, for example, starting from whole numbers,* (https://en.wikipedia.org/wiki/Natural_number) *then defining rationalnumbers algebraically* (https://en.

fractional number
fraction
root number
root

wikipedia.org/wiki/Rational_number), *and finally defining realnumbers as equivalence classes of their Cauchy sequences* or (*) *as Dedekind cuts, which are certain subsets of rational numbers.*" (https://en.wikipedia.org/wiki/Real_number#Definition)
(*) This is in fact incorrect: one does *not* have a *choice* between the Dedekind route and the Cauchy route and one should *both*:
i. Go the Dedekind route *and* extend the *metric* and then prove that the quotient is *metric*-complete,
and then
ii. Go the Cauchy route *and* extend the *order* and then prove that the quotient is *order*-complete,
and finally
iii. Prove that the two quotients are both *metric*-isomorphic and *order*-isomorphic.

Which, unless you are a mathematician, is not exactly enlightening either. In any case, a very*, very tall order.*

2. Fractions and roots. Originally, **fractional numbers**, **fraction** for short, were the numbers with which to denote amounts of stuff.

Example B.1. $+\frac{3}{4}$ **Gallon of milk**

But, to begin with, *defining* fractional numbers is not that simple and then a fraction is only like a Birth Certificate in that a fraction is just a *name* that says where the fraction is coming from and a fraction certainly does *not* provide any indication of what the size of the fraction might be.

Example B.2. The *fraction* $\frac{4168}{703}$ is just a *name* for the solution of the equation $703\,x = 4168$ (Assuming the equation *has* a solution!)
And, up front, it is certainly not clear how $\frac{4168}{703}$ compares with, say, $\frac{4167}{702}$ or even with $\frac{4}{7}$

And then it was realized that not every amount of stuff could be described by a fractional number of a given unit of stuff.

Example B.3. Take the side of a square as unit of length. Then the diagonal of the square is *not* a fractional number of the side. (https://en.wikipedia.org/wiki/Irrational_number.)

So, **root numbers**, **root** for short, were invented but again a root is just a *name* that says where the root is coming from but a root certainly does *not* provide by itself any indication of what the size of the root might be.

EXAMPLE B.4. The *root* $\sqrt[3]{+17.3}$ is just a *name* for the solution of the equation $x^3 = +17.3$. (Assuming the equation *has* a solution!)
And up front, it is certainly not clear how $\sqrt[3]{+17.3}$ compares with, say, $\sqrt[2]{+18.5}$

And, worse, fractions and roots are *best* cases and most real numbers do not tell us where they are coming from and even less how to get even a rough idea of what the size of that real number might be.

You just have to find out from somewhere.
In textbooks it's of course the other way around,

EXAMPLE B.5.

- π is just a *name* that does *not* say by itself that π is "*the ratio of a circle's circumference to its diameter*". (https://en.wikipedia.org/wiki/Pi)
- e is just a *name* that does *not* say by itself that e is "*a mathematical constant which appears in many different settings throughout mathematics*". (https://en.wikipedia.org/wiki/E_(mathematical_constant))

*While CALCULUS goes all the way back to the late 1600s (**??** (**??** **??**, **??**)), DISCRETE MATHEMATICS goes only back, at the very earliest, to the early 1900s.*

OKsoFAR **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR** **OKsoFAR**

2 Calculating with real numbers.

1. Title. This can be done directly from the names *only* with the same two kinds of real numbers, that is when the real numbers are fractions or roots:

i. When the real numbers are fractions, there are procedure to compare, add, subtract, multiply and divide directly from the whole numbers that make up the fractions. (https://en.wikipedia.org/wiki/Rational_number#Arithmetic) decimal approximation

Example B.6. To know which is the larger of $\frac{4168}{703}$ and $\frac{5167}{831}$ there is a procedure that involves only the wholenumbers 4168, 703, 5167 and 831 namely $\frac{4168}{703} < \frac{5167}{831}$ if and only if $4168 \times 831 < 703 \times 5157$.

ii. When the real numbers are roots, there are procedures to multiply and divide directly with the whole numbers that make up the roots but *not* to add or subtract. (https://en.wikipedia.org/wiki/Nth_root#Identities_and_properties)

Example B.7. $\sqrt[2]{5} \times \sqrt[3]{7} = \sqrt[2\times 3]{5^3 \times 7^2}$

iii. However, it is usually not possible to calculate with both kinds of real numbers at the same time.

Example B.8. Add e and π and/or figure out which of the two is larger. (Hint: you can't do either from the names.)

And, even when the real numbers are fractions and roots, things can still be difficult.

Example B.9. Add $\sqrt[3]{64}$ and $\frac{876}{12}$ and/or figure out which of the two is larger. (Hint: in *this* case you *can* do both but *not* in the only slightly different case of $\sqrt[3]{65}$ and $\frac{875}{12}$.)

And at the expense of forcing memorization of scattered recipes.

iv. Of course, the examples in textbools use mostly fractions and/or roots even though it is at the expense of being immensely misleading if only because *most* real numbers are *neither* fractions *nor* roots.

3 Approximating Real Numbers

The reason *engineers* and *physicists*, *chemists*, *biologists*, don't worry about real numbers is because about the first thing they do is to replace real numbers by **decimal approximations**, that is . . . signed decimal numbers!!!

1. Approximating. To begin with, one way or the other, *all* real numbers, *including* fractions and roots, come with a **Procedure** for calculating approximations by numbers.

i. To approximate fractions, we use the division procedure.

EXAMPLE B.10. To approximate $\frac{4168}{703}$, we *divide* 703 into 4168.

Few divisions end by themselves. Fortunately, though, when they don't, the more we push the division, the better the approximation.

ii. To approximate roots, we essentially proceed by trial and error.

EXAMPLE B.11. To approximate $\sqrt[3]{17.3}$, we go:

- $1.0^3 = 1.0$
- $2.0^3 = 8.0$
- $3.0^3 = 27.0$,

Since 17.3 is between 8.0 and 27.0, $\sqrt[3]{17.3}$ *must* be somewhere between 2.0 and 3.0. (But *how* do we know that it *must*?) So now we go:

- $2.1^3 = 9.261$

  ...
- $2.5^3 = 15.620$
- $2.6^3 = 17.576$

Since 17.3 is between 15.620 and 17.576, $\sqrt[3]{17.3}$ *must* be between 2.5 and 2.6. (But *how* do we know that it *must*?)

And so on. (The actual procedure is more *efficient* but that's the idea.)

Of course, the more "exotic" the real number is, the more complicated the procedure for approximating is going to be:

EXAMPLE B.12. There are many ways to approximate π. The simplest one is the Gregory-Leibniz series whose first few terms are:
$\frac{4}{1} - \frac{4}{3} + \frac{4}{5} - \frac{4}{7} + \frac{4}{9} - \frac{4}{11} + \frac{4}{13} \ \dots$
However, even with "*500,000 terms, it produces only five correct decimal digits of π*" (https://en.wikipedia.org/wiki/Pi#Approximate_value) But there are shorter if more complicated ways to approximate π.

EXAMPLE B.13. One of the very many ways to approximate e is:
$1 + \frac{1}{1} + \frac{1}{1\cdot 2} + \frac{1}{1\cdot 2\cdot 3} + \frac{1}{1\cdot 2\cdot 3\cdot 4} \ \dots$
(https://en.wikipedia.org/wiki/E_(mathematical_constant)#Asymptotics)

2. Approximation error. Since a real number is usually *not* equal to the signed decimal number used to approximate it, in order to write

[...]
largest permissible error

equalities we will have to use:

DEFINITION B.1will be the symbol for "some infinitesimal number, positive or negative, whose size is too small to matter here".

In other words, [...] is a *signed* number about which the only thing we know is that the size of [...] is *less* than the **largest permissible error** whichi is the equivalent here of a tolerance.

EXAMPLE B.14.

- $\frac{4168}{703} = 5.929 +$ [...] where [...] is less than 0.001 which is the largest permissible error. (Else the procedure would have generated 5.928 or 5.930 instead of 5.929.)
- $\sqrt[3]{17.3} = 2.586\,318\,666\,944\,673 +$ [...] where [...] is less than $0.000\,000\,000\,000\,001$ which is the largest permissible error. (Else the procedure would have generated $2.586\,318\,666\,944\,672$ or $2.586\,318\,666\,944\,674$ instead of $2.586\,318\,666\,944\,673$.)
- $\pi = 3.141\,5 +$ [...] where [...] is less than $0.000\,01$ which is the largest permissible error. (Else the procedure would have generated $3.141\,4$ or $3.141\,6$ instead of $3.141\,5$.)
- $e = 2.718\,281\,82+$[...] where [...] is less than $0.000\,000\,01$ which is the largest permissible error. (Else the procedure would have generated $2.718\,281\,81$ or $2.718\,281\,83$ instead of $2.718\,281\,82$.)

4 The *Real* Real Numbers Are The *Regular Numbers*

xxxxxxxxxxxx

1. Title. So, "*the wheel is come full circle*" (King Lear), from the real numbers all the way back to the real world numbers, with just one question left:

And a good question it is. But then, the answer surely depends on what you mean by "learn".

Why should people who want to learn CALCULUS have to use real numbers which they would eventually have to *approximate* with ... real world numbers anyway?

But since,

▸ To complete the quote from Gowers in Subsection 1.3 - Whole numbers vs. decimal numbers (Page 5), "*Physical measurements* are not real numbers. *That is, a measurement of a physical quantity will ...*"

number
real world number

and

▸ Just like people, "*most calculators do* not *operate on real numbers. Instead, they work with finite-precision* [decimal] *approximations.*"(https://en.wikipedia.org/wiki/Real_number#In_computation.)

the answer must surely be, as *Engineers* used to be fond of saying, that:

And even if it is eventually to become a mathematician that you want to learn CALCULUS, as Timothy Gowers said, "There is nothing wrong with thinking of real numbers as signed decimal numbers with infinitely many decimals: indeed, many of the traditional arguments of analysis become more intuitive when one does." [1]

"The ***real* real numbers** are the signed decimal numbers."

=======Begin HOLDING=======

So, in view of the fact that we will use No other number (CAUTION 0.2, Page 5) than signed decimal numbers and since always having to write the qualifiers "signed decimal" to qualify the word "number" would be unbearably burdensome:

AGREEMENT B.1 In the absence of qualifier, in *this* text the word **number** will *always* be short for signed decimal number.

Money talks so, even in the non-metric USA, banks now use signed decimal numbers in their accounting.

EXAMPLE B.15. What we will intend by:

- "Numbers are beautiful" is "*Signed decimal* numbers are beautiful",
- "Plain numbers are cute" is "Plain numbers, *whether whole or decimal*, are cute".
- "Decimal numbers are handsome" is "Decimal numbers, *whether plain or signed*, are handsome".

=======End HOLDING =======
=======Begin HOLDING=======

2. Real world numbers. So, like all *scientists* and *engineers*, the numbers *we* will use will be

DEFINITION B.2 Real world numbers are (signed decimal) numbers all whose digits are significant.

[1] https://www.dpmms.cam.ac.uk/~wtg10/decimals.html

And real world numbers are not at all the same as 'Real Numbers' which will be discussed in ?? ?? - ?? (??)

real world number

And so, from now on,

AGREEMENT B.1 (Restated) 'Number' (without qualifier) will be short for real world number.

=======End HOLDING =======

relative

Appendix C

Localization

Inputs are counted from the origin that comes with the ruler. However, rather than counting inputs **relative** to the origin of the ruler, it is often desirable to use some other origin to count inputs from.

Appendix D

Equations - Inequations

The following is essentially lifted from REASONABLE BASIC ALGEBRA, by *A. Schremmer*, freely downloadable as PDF from (Links live as of 2020-12-31):

- Lulu.com (https://www.lulu.com/en/us/shop/alain-schremmer/reasonable-basic-algebra/ebook/product-1m48r4p5.html?page=1&pageSize=4)

and/or

- ResearchGate.net (https://www.researchgate.net/publication/346084126_Reasonable_Basic_Algebra_Lulu_2009)

Appendix E

Addition Formulas

1 Dimension $n = 2$: $(x_0 + h)^2$ (Squares)

In order to get

Appendix F

Polynomial Divisions

1 Division in Descending Exponents

Since *decimal numbers* are combinations of powers of TEN, it should not be surprising that the procedure for dividing decimal numbers should also work for *polynomials* which are combinations of powers of x.

259

Appendix G

Systems of Two First Degree Equations in Two Unknowns

1 General case

xxxx xxxxx xxxxx

Appendix H

List of Agreements

Appendix I

List of Cautionary Notes

Appendix J

List of Definitions

Appendix K

List of Language Notes

Appendix L

List of Theorems

Appendix M

List of Procedures

Appendix N

List of Demos

Index