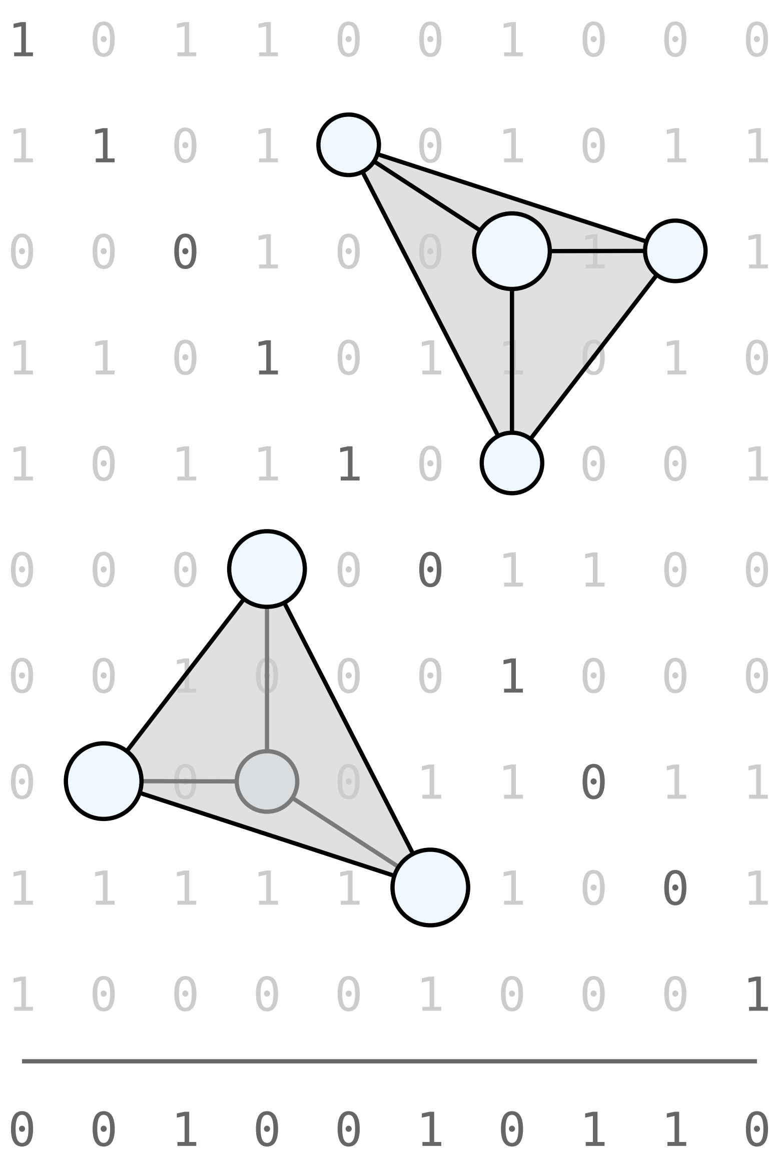
INTUITION FOR COMPUTATION

A Guide for Programmers and Curious Minds



Tommy Monson

CONTENTS

Preface

I wrote my first novel because I wanted to read it.

—Toni Morrison

In December of 2018, I graduated from Duke University, feeling like I knew simultaneously very much and nothing at all.

Acknowledgements

The very word intuition has to be understood.

You know the word tuition—tuition comes from outside, somebody teaches you, the tutor.

Intuition means something that arises within your being;

it is your potential, that's why it is called intuition.

Wisdom is never borrowed, and that which is borrowed is never wisdom.

Unless you have your own wisdom, your own vision,

your own clarity, your own eyes to see,

you will not be able to understand the mystery of existence.

—Osho

Philosophy of Computation

> *We're presently in the midst of a third intellectual revolution. The first came with Newton: the planets obey physical laws. The second came with Darwin: biology obeys genetic laws. In today's third revolution, we're coming to realize that even minds and societies emerge from interacting laws that can be regarded as computations.* ***Everything is a computation.***
>
> —Rudy Rucker

Computation is an essential part of being human. Just as we act on our perceptions, feel our emotions, and daydream in our imaginations, we also compute with our *reasoning* in order to find answers to the questions we have about life.

One can describe computation in a variety of ways. Some are poetic, others are more formal, and many we will discuss at length. In doing so, we will generally flow from poetry to formality, starting with broad statements and colorful examples and progressing toward a formal understanding. For now, we will say that computation is a process that resolves *uncertainty*.

In his *Theogony*, the Ancient Greek poet Hesiod (fl. 750 BCE) depicts a fascinating uncertainty known as Chaos: a vast nothingness that preceded the creation of the Universe. From this void emerged the primordial deities, personifications of nature who gave life to the Titans and, by extension, to the Olympian gods. Curiously, the modern definition of *chaos* has nearly inverted. Now, one might say that a situation is *chaotic* if there is *too much* going on. Despite their great conceptual difference, these meanings share a common property: they both evoke *confusion*.

This place in which we find ourselves may indeed be post-Chaos, but confusion remains a familiar state for us. In fact, it is our natural state. We come into life confused about everything—crying, open-eyed in reaction to how much is going on around us. Slowly, we figure a few things out. Later, we consider many things *certain*. As time ticks on and we carve our separate paths in the world, the wise ones among us come to realize the true depth of their confusion.

Although we are initially confused, we naturally pursue an understanding of the chaos that surrounds us. We perceive the goings-on of our environment and draw *information* from them, piecing together a *knowledge* that orients us and gives meaning to the disorder of our existence. This is the crux of computation. It is the structured manipulation of *data* into *information* for the purpose of acquiring *knowledge*. It shines a light into the darkness we call home.

These terms—data, information, and knowledge—are broad enough to apply to a variety of systems. In fact, they appear in nearly every academic field and certainly within those that strive for objectivity. One can have data on, information about, and knowledge of anything that can be observed or experienced. As such, it is difficult to confine these terms to short and tidy definitions that are both rigorous and free of controversy. We will have to settle then for long, pragmatic definitions.

In this chapter, we will explore what computation is and why it matters. In some sense, computation is a kind of game in which *pieces* are moved and transformed within a *game space* according to a predetermined set of *rules*. However, unlike most games, computation has no set goal. Nature shuffles its hazy pieces about, following arbitrary rules for reasons we cannot ascertain. Man, on the other hand, explicitly defines its pieces and rules, modeling them after Nature's strange mechanics. We invent *win conditions* and play toward them, hoping that our computations will result in general solutions that will elucidate the knowledge we seek.

SECTION 1: *Information and Communication*, covers the pieces ...

SECTION 2: *Reasoning and Logic*, covers the rules ...

Here, we learn how this game is played and how it is woven into the fabric of our reality. In the chapter following this one, we model the game of computation mathematically and find the theoretical limits of what it can tell us. Beyond that, we learn how to be an excellent player, covering *techniques* for efficient problem solving and *tools* for designing beautiful, maintainable systems.

1 INFORMATION AND COMMUNICATION

> *For Wiener, entropy was a measure of disorder; for Shannon, of uncertainty. Fundamentally, as they were realizing, these were the same.*
>
> —James Gleick

The most general concepts are among the hardest to grasp, and *information* is about as general as you can get. Potentially *everything* can be described in terms of information—the physical world, all of literature and written language, and perhaps even our conscious experience. Thus, our path to a proper understanding of information and how it relates to reality will be an arduous one, requiring an open mind and an ample store of time set aside for contemplation. But it will also be an elegant one, weaving its way through philosophy and physics and arriving at an *information theory* that we can use to explain the behavior of engineered systems, human language, and the intersection of the two: mechanical computation.

You already have an idea of what information is, and it is likely that your intuition is broadly correct: information is a thing that tells you something. As we dive into more and more abstract topics, do not hesitate to, in this way, fall back on your existing vocabulary for support. A word used technically is more formal and precise than the same word used casually, but the gist of it all is often the same.

Recall the usual context in which technical language is born. A phenomenon is observed by a community, and it becomes the subject of casual discourse. It may even be a widespread concept that is discussed by society at large, in some way or another. The phenomenon is accordingly given an *informal definition* and a name (or, perhaps, many names). Interested theoreticians then study the structure of the phenomenon and *formalize* it. They give it a *formal definition* and bestow it a more permanent name, either one used previously or something new that is, in the namers' opinion, more apt. In the latter case, the new term is disseminated throughout the community and perhaps to the wider public, where it may either replace or coexist with previous informal terms.

Regardless, the namers must reach outside of the formalisms they have constructed and find a word from an informal language that captures the essence of their abstract thought. This is the beauty of good jargon: such a word will, to an amateur, shed a shallow but wide light into the vast space of an idea, but it will also illuminate the whole shebang in the eyes of an expert, evoking in its concise informality all that he or she has previously learned on the subject.

In the spirit of this principle, four words are given below, defined softly, that will act now as beacons in the dark of our ignorance and later as summaries of our hard-earned understanding:

> ***Data*** *are pieces of information that are meaningless when considered separately. They are observable lacks of uniformity in reality, each of which may be represented by a symbol that conveys its particular difference from the norm.*
>
> ***Information*** *is an ordered arrangement of data that is considered meaningful. The data are arranged such that the information as a whole complies with the grammar of a language that can, in theory, be understood.*
>
> ***Knowledge*** *is an understanding of information that emerges given proper interpretation and thought. It is usually associated with Truth and may be considered a body of true beliefs.*
>
> ***Wisdom*** *is the capacity to think and act in a way that is right. It involves the use of knowledge, a prudent choice of means and ends, and a healthy doubt of one's own understanding. It is born of an acknowledgment of how little one truly knows and how little control one truly has.*

With these fundamental terms in mind, let us take a page from Hesiod and begin our tale with an uncertain chaos. Let us regard reality as a baby might: confusing but ultimately intelligible through *categorization*.

1.1 Categories of Being

Dark turns into light. Blurry figures appear. You see. You hear. You *feel*. The knife-edge of experience materializes, and you step out onto it. There is no thought—only sensation. Air kissing skin and turbulent color. Sonic frenzy and the feeling of self-weight. Reality emerges in its raw, incoherent form, and nothing is gleaned—you are lost in it all. Indifferent, this phantasmagoria rages on.

This is roughly the experience of a newborn infant. For children of thirty days or less, life is senseless sensory mayhem. They are not conscious, at least not in the way that we typically think of consciousness. They are, however, aware of their surroundings and sensitive to *phenomena* (i.e. those things which *appear* in reality, *observable* things). To newborns, the world is not a system of distinct parts but an all-encompassing soup of stimuli. Their sensory organs input data that they do not understand, and they then output *reflexes* that have been programmed into their biological firmware.

This is not to say that babies are mindless. Rather, their minds are just unlike those of typical adults, which both *filter* and *store* information. For example, while babies and adults have similar hearing ability, babies nevertheless hear things that adults do not. This is because adults subconsciously choose to ignore certain sounds; they may *sense* the auditory data, but they do not *perceive* them. This filtering process is called *sensory gating*, and it inhibits any stimulus that is deemed irrelevant. Any information that is not filtered is then eligible for storage in *memory*.

Thus, if the adult mind is like the conical beam of a flashlight, seeing far but neglecting much, the baby mind is like the radiant orb of a lantern, seeing everything nearby but illuminating nothing deeper. These configurations are well-suited to the goals of each: the adult needs *attention* to understand complex situations whereas the baby needs *unrestricted perception* to acquire as much information as possible. Unburdened by categorical thought and sensory gating, babies live in the present and see the *suchness* of reality. However, their experience is not quite the *kenshō* of Zen—it is more like Aldous Huxley's Mind at Large or William Blake's famous metaphor, from which Huxley took the title of his essay on psychonautics:

> *If the doors of perception were cleansed every thing would appear to man as it is: Infinite. For man has closed himself up, till he sees all things thro' narrow chinks of his cavern.*

So it is through a great deal of concurrent filtering and learning that we come to build a cavern around ourselves. But we do not muffle our perception pointlessly—we trade it for genuine *thought*.

1.1.1 *Category Systems*

There is a distinction that is sometimes made in cognitive science between a *(phenomenal) P-consciousness* and an *(access) A-consciousness*, the former being the sole consciousness of the baby mind, concerned only with bits of immediate, subjective experience known as *qualia* (e.g. what it is like to *feel* a delicate raspberry as it touches your tongue *at this moment*, what it is like to taste it as you mash it between your molars, the experience of seeing the reds and greens of this *particular* berry bush *here and now*, etc.) and the latter being the dominant consciousness of the adult mind, concerned with *neural information* (e.g. thoughts about sensory data, memories, abstract ideas, plans for the future, and all things involved in *mental computation*).

Of course, adults still have a P-consciousness—we can lose ourselves in the moment if we are willing. But ours is a boarded-up P-consciousness, largely unaware of the immense volume of all that is happening at all times. Perhaps it is only the bodhisattvas or the mystics of yore who truly come to know their infant minds again and see the P and A as they are: two *modes* of a unified whole. Or perhaps, more pessimistically (and as

Huxley suggests), the P in its unadulterated totality is really a kind of schizophrenic insanity, and it is only by the grace of our specialized neural hardware that we erect defenses against its sensory onslaught. Alas, the suchness of such a consciousness eludes those of us with at least one foot planted solidly on the earth.

The Hard Problem of Consciousness

But really, what is the nature of P-consciousness? Can it be studied scientifically? As this consciousness is the realm of qualia, the answers depend on what we understand qualia to be.

In any event, qualia are beyond our scope.

1.1.2 *Objects*

This curiosity aside, we could also design our category system such that quantities and qualities are both considered *properties* of some very general sort of thing. Philosophers have long considered the idea of a highest-level category and have ascribed a variety of names to its members: *entity*, *being*, *thing*, *term*, *individual*, etc. Each carries its own subtle connotations. We will use the name *object* because it is the default choice in current philosophical practice and because it has bled into the technical language of computer science with its meaning largely unchanged.

How general is this category of objects? Namely, are properties objects? If they are, then our system collapses to a single category, which really means that no distinctions about reality can be made at all. The category of objects thus becomes the soupy mess we are trying to escape. For now, let us assume that properties are *not* objects. Instead, they are something else, and they *relate* to objects in some way, giving them their characteristics. From these relations, the diversity of existence unfolds.

The **abstract-concrete dichotomy** classifies *things* as being either *abstract* or *concrete*. Despite the fact that most people recognize a difference between an abstract and a concrete thing, this dichotomy does not have a universally accepted definition. That said, being abstract is often defined as being both *non-spatiotemporal* and *causally inefficacious*. That is, something is abstract if and only if

1. it does not exist in *space* or in *time* (or, alternatively, in *spacetime*), and
2. it cannot be the *cause* of any *effect*.

However, one might argue that nothing is truly causally inefficacious. Indeed, if something was truly unable to cause *anything*, it would not be able to affect the *thoughts* of any philosopher, mathematician, or scientist for whom such an abstract thing would be of interest. And yet *mathematical objects* (e.g. a number, a line, a cube, the sine function, etc.) are often considered abstract, despite the fact that they affect the thoughts of mathematicians daily, thoughts which in turn prompt concrete behavior, such as drawing diagrams. Perhaps, causal inefficacy is too strict a requirement.

Let us consider why one might want to refer to something that is abstract. An abstract object references *properties* that many concrete, causally efficacious objects have. For example, a sphere does not exist in space or time; it is *nowhere* and *never*. It also does not seem capable of causing an effect in the way that, say, a ping-pong ball can. And while the abstract sphere may not have the same sort of causal efficacy as concrete spherical objects, it does have a *relation* to those objects. That is, the concept of sphericality is *related* to the *class* of concrete objects that may be reasonably modeled as spheres. Thus, an abstract object is, at least, convenient because it can identify a class of concrete objects that is of interest by specifying relevant properties.

In light of recent advancements in fields like neuroscience and artificial intelligence, many believe that the human mind is nothing more than a physical system, albeit a particularly complicated one, that can be modeled mathematically like any other. So it may be that abstract objects are just interpretations of concrete electrochemical signals in our brains. Regardless,

1.1.3 *Properties*

A common approach to this challenge is creating a system of *categories* that partition reality at the highest level. For example, one could suggest that there is a fundamental difference between a *quantity* and a *quality*. If these are considered distinct categories, we can classify *measurable* or *countable* stuff like *length*, *mass*, *number*, and *monetary value* as quantitative and *experiential* stuff like *softness*, *roundness*, *flavor*, and *beauty* as qualitative.

The quantity-quality distinction is a generally accepted one. However, the dichotomy is not so absolute. For example, softness is listed above as a quality. We might expect, then, that hardness is a quality. For example, rocks are hard and puppy ears are soft. But hardness is also a quantitative measurement of resistance to plastic deformation. Thus, the word *hardness* can refer to both quantitative and qualitative phenomena. The categorical difference between these hardnesses, which exists independent of language, is identified through the context in which the word is used. A hardness can be measured with numbers, but it can also simply be *experienced* via touch.

Space and time are two features of our reality. As they appear to us, the former is an expanse in three dimensions, and the latter can be thought of as an arrow that extends steadily in one dimension, from past to future. They are inextricably woven into the fabric of our experience, and yet their true nature eludes us.

Consider a space that contains many particles of *substance*, all of which move about constantly as time goes on. Suppose that, one by one and randomly, these particles blink out of existence forever. When only one particle remains, is that which is *not* the particle still considered space? Further, can we still say that this particle moves? It remains surrounded by a void, interacting and relating with nothing at all, participating in no distinguishable events. And if indeed nothing changes, does time still tick? If so, will space and time *mean* anything after that last, lonely bit of matter disappears and everything becomes a uniform non-existence?

These are the questions that pertain to whether space and time are *absolute* or *relative*. If space is absolute, like a fixed three-dimensional grid that pervades the material realm, we might ask where the origin $(0, 0, 0)$ is. If such a location were to exist, every object and particle in the Universe would have an absolute position in three coordinates. In other words, position would be a distinguishing feature of each and every object, and it would not be defined in relation to other objects, but to this primordial *center of the Universe*.

If space is instead relative, a conceptual origin can be placed wherever one likes. Distances can then be measured from this point. Of course, this can be done in an absolute space as well. Thus, the absolutist viewpoint commits the cardinal sins of being *unobservable* and being *useless*. To any human observer, a reality with absolute space would look no different than a reality with relative space—objects would not change in any way after being related to an arbitrary center. Access to these absolute positions also would not provide us with any additional information—they would just be fixed offsets of any positions measured from a relative point.

Perhaps then, space is is better thought of as *relational* rather than *objective*. Perhaps it is not a hidden grid-like object itself, but an *ordering* that is composed of relations between objects. And perhaps then it is also better thought of as a product of the *mind* rather than as a physical thing. That is, space may instead be an *abstraction* that our minds make by estimating the relative distance between two objects and comparing it to every other potential spatial relation in our surroundings. In this view, space is simply a web of relations that is constructed by a sentient being.

Similarly, time may not be a hidden clock-like object, but instead a series of relative temporal intervals between *events*. And if we only know that time passes because of the changes we perceive in space, it is possible that time is an abstraction as well, an ordering of events made by a sufficiently conscious mind. Furthermore, if space and time are both relational and coupled, perhaps it is better to think of them as a unified *spacetime* despite our predispositions to think of them separately.

The philosophy of space and time is full of eternal questions about infinity, substance, void, uniformity, asymmetry, and consciousness. Comparatively, *computational space and time* are simple.

1.1.4 *Classes*

A **domain of discourse** is a set of all the things under discussion. It is also called a *universe*, particularly in the field of mathematical logic. It specifies, out of all conceivable objects of study, those objects which are pertinent to the matter at hand. It gives *context* to what we say.

A *discourse* is a conversation, and its *domain* is the subject of the conversation. A domain may also refer to an entire field of study, perhaps one whose conversation has been developing for thousands of years (e.g. physics, whose domain is physical reality). Each field of study has an accompanying domain of discourse, the set of objects within its purview. Working in such a field is thus akin to exploring a *universe* populated with such objects.

As we observe the objects of a universe, we can make *logical statements* about them. We can also ensure our statements are consistent by using a *formal system of deduction* to prove them from a set of *axioms*. And if we compile every statement that can be proven in this system—the axioms and their consequences—we will form a *theory*, a set of true statements (or *theorems*) that describes the universe in question.

The word *theory* may also refer to a *theory-in-progress*, a body of logical work to which theoreticians contribute. We might consider this an *open theory*, a theory for which we do not have all possible knowledge. In contrast, a *closed theory* is one for which we know everything there is to know.

Similarly, we can describe universes as open or closed. The *open-world assumption* is made when we do not have complete knowledge of a system: if a statement is not known to be true (i.e. it is not a member of our theory), we do not assume anything about its truth value. This is the case for any system that is *discovered*, like the various systems of nature.

Computational systems, however, are not discovered but *designed*. Under normal conditions, it is possible to know everything that happens during a digital computation—all of the information appears in a discrete, finite space of computer memory. Thus, we are omniscient with regard to this universe and should make the *closed-world assumption*: if a statement is not known to be true, it must be false.

The **type-token distinction**

The Universal and Particular Notions of Chess

As an illustrative example, consider all of the games of chess that are being played right now, currently progressing or left half-finished and awaiting the next move. Consider also all of the *historical* games of chess that have been played to completion or total abandonment since the invention of chess.

1.1.5 *Relations*

Identity seems to be a mandatory quirk of being. It is difficult to imagine an existence in which things are not themselves. And yet, it is also difficult to define, in any meaningful way, this supposedly crucial concept.

The **map-territory relation**

1.1.6 *Ontologies*

Say you want to *communicate* an idea X to someone. That is, you would like to convey some *information* to them so that you might together establish a shared body of *knowledge*. X is on the tip of your tongue, but you cannot quite put it into words. Accordingly, you express a related idea Y_0, hoping that you might be able to suss out your target X with the help of your fellow interlocutor. This proxy idea is either too specific or too general, a *subclass* or *superclass* of the *class* X, and you must replace it with another idea, preferably X or at least some Y_1 that is closer to X than Y_0. And so on and so forth, ideas are proposed until some candidate $Y_n = X$ is found. The conversation could go as follows:

"Dude, what's that thing with the moving parts that beeps sometimes? And it, like, gets hot if it moves too fast?

. . .

It's like a *Machine* of some sort, but more *specific*."

"Oh yeah, those expensive things with the metal chassis and all the buttons? About 15 feet long with two rows of seats?

Sounds like a good ol' fashioned *Sedan*, my dude."

"Nah, it's not a sedan. That's close, but I'm thinking of something more *general* than that. It's definitely a *type* of *Vehicle* though."

"Well, there are a lot of *Vehicle* types out there, but I'm guessing this has to do with a *Car*, specifically my—"

"Of course! I was thinking of a *Car*.

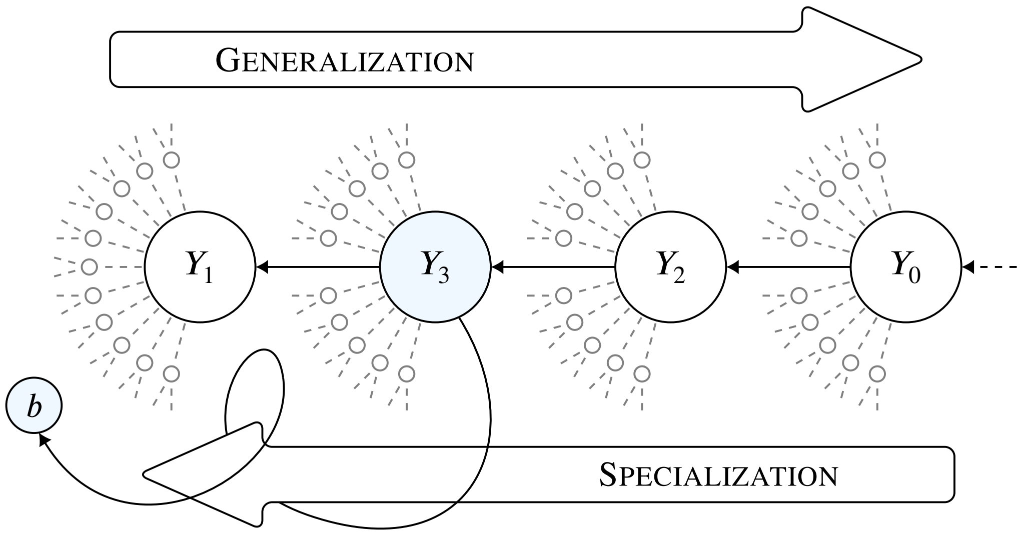
Ok, now that we're on the same page, about *your car*...
My bad, man. I crashed it into the lake last night."

. . .

"Hey, if it makes you feel better, Joey dared me to do it, so there wasn't really an alternative."

In this scenario, Interlocutor $\mathcal{A}$ introduces the problem: there is *information* to *communicate*, and it defines a *class* X. And a *class*, we understand, is simultaneously a collection of individual *objects* and an *abstract object* itself, a formal definition of the collection to which all members comply. $\mathcal{A}$ grasps for X, throwing out descriptions to lure it closer, but it evades. $\mathcal{A}$ then takes the class Y_0, *Machine*, as a starting point and states that it is a *superclass* of X.

Interlocutor $\mathcal{B}$ responds with *specialization*, adding assumed details to the definition of Y_0. Thus, Y_1 is proposed: the class *Sedan*. Luckily, the assumptions are not totally off-base; Y_1 turns out to be a *subclass* of X. $\mathcal{A}$ confirms this and so returns with a *generalization* that conjures Y_2, the class *Vehicle*. $\mathcal{B}$ *specializes* again, offering up as Y_3 a subclass of *Vehicle* called *Car*. Finally, $\mathcal{A}$ accepts this class as the target X and ends the search. And with this mutual knowledge of *Car* established, $\mathcal{A}$ then refers to an *instance* of *Car*, a particular car b that belongs to $\mathcal{B}$. And with both interlocutors now cognizant of exactly what is being discussed, $\mathcal{A}$ swiftly informs $\mathcal{B}$ that his ride is totaled.



In the above diagram, we see the Y classes and their relations to each other. The arrows between them are *subsumption* relations. They point from a *parent* class P to a *child* class C, and we say that P *subsumes* C (i.e. it encompasses the entirety of C). In set theory notation, $P \subset C$. Alternatively, we can reverse the arrows and call them instead *is-a* relations, which point from C to P. Every $c \in C$ is a member of the more general class P, and thus every c **is a** P of some *type*.

Of course, any given type T can have many *subtypes*, and those subtypes can have their own subtypes. A subtype of T builds on the information that defines T, adding new information that specializes the underlying parent class. And with enough specialization, these subtypes may become quite unlike each other at a glance, sharing little else than a distant relation to T. This branching nature of subsumption is depicted above in gray.

Like branches that furcate from an old oak tree and fork into smaller branches toward the leaves, types spread in a *recursive* manner, each one potentially a parent of arbitrarily many children and an ancestor of arbitrarily long lineages. As a *type system* like this grows deeper and denser, more information is required to represent it. Similarly, any class in the system *gains information* as it is specialized and *loses information* as it is generalized. Thus, a class of type T is *also* of any supertype of T because each possible supertype represents a subset of T's information. For example, the class *Car* is of type *Car*, but it is also of the more general types *Vehicle* and *Machine*.

Naturally, one wonders about what might happen at the extremes of generalization and specialization. Three relevant question-answer pairs are given below:

- *Is there a type so general that it describes everything?*

  Indeed, there is: it is the *top type* $\top$, and it describes a special class known as a *domain of discourse* or *universe*. The names speak for themselves—the supertype of *everything* describes the class of *all things* under discussion, the *universe* of relevant topics, the *domain* of our discourse. $\top$ serves as a useful label in situations when type is unimportant or indeterminate.

- *Is there a type so specific that it describes … nothing?*

  Yes. It is the *bottom type* $\perp$, and it describes the *empty class*, the class with zero members. It can be considered the type of an *overspecified* class. That is, the class has been defined such that no object in the *domain of discourse* complies. $\perp$ describes things that are *out of scope* or *nonexistent*. Thus, it is often used as the return type of functions that *diverge* (i.e. do not terminate properly) or as the type of *nothing* (e.g. the 5^{th} element in a 4-element list is nothing).

- *Are instances just classes that are specialized until they have one element?*

  Technically, no. Specialization is a class-to-class process: a class is specialized to a subclass. *Instantiation*, the construction of a concrete *instance* from an abstract description, is instead a class-to-object process. For example, in the above diagram, the class Y_3 is *instantiated* in order to produce the object b (or, by reversing the direction of the arrow, we may say that b is an **instance of Y_3**).

  Because a type describes a class, it is also incorrect to say that b defines its own subtype of *Car*. Rather, b is a *token* of type *Car*, a physical object that *realizes* one of the many possible designs that fall under the class *Car*. However, Y_3 could be specialized such that b is the sole element of the resultant class b. A class with one member is called a *singleton class*, and it has a *unit type*, a type with one valid value. The type of b would be a subtype of *Car* that describes only b, but neither b nor its type are the same thing as the *instance b*.

1.2 Classical Metaphysics

Blah

1.2.1 *Henosis and Arche*

Blah

1.2.2 *Democritus' Atomism*

Blah

1.2.3 *Plato's Theory of Forms*

Plato (c. 428/427 – c. 348/347 BCE) was the first to expound a metaphysics of objects with his *theory of Forms*, which proposed a solution to the *problem of universals*. This problem is subtle, but important: how can one general property appear in many individual objects? For example, when we say that a floor and a bowl are *smooth*, are we referring to some singular paradigm of *smoothness* that is independent of each? For Plato, *universals* or *Forms*, such as smoothness, are indeed distinct from the *particulars* that *partake* of them, such as floors and bowls and other smooth objects. He posits that the *essence* of an object, its *sine qua non*, is the result of its partaking of these Forms.

Forms, as conceptualized by Plato, are ideal, abstract entities that exist in the *hyperouranos*, the realm beyond the *heavens*. In the times of Homer and Hesiod, the Ancient

Greeks understood a cosmology that closely mirrored that of the Hebrew Bible. They envisioned the Earth as a flat disc surrounded by endless ocean and the sky as a solid dome with gem-like stars embedded in it, much as it all appears from a naive perspective on the ground. Under the Earth was the *underworld*, where the dead live, and beyond the dome were the *heavens*, where the divine live.

Plato, however, was influenced by the philosophy of Pythagoras (c. 570 – c. 495 BCE), who championed a spherical Earth for aesthetic reasons. Thus, he instead modeled the Earth as a "round body in the center of the heavens," and he imagined the Forms as part of a different, eternal world that transcends physical space and time. Whereas physical reality is the domain of perception and opinion, the realm of the Forms is, as he paints it in *Phaedrus*, the domain of *reason*:

> *Of that place beyond the heavens none of our earthly poets has yet sung, and none shall sing worthily. But this is the manner of it, for assuredly we must be bold to speak what is true, above all when our discourse is upon truth. It is there that true Being dwells, without colour or shape, that cannot be touched; reason alone, the soul's pilot, can behold it, and all true knowledge is knowledge thereof. Now even as the mind of a god is nourished by reason and knowledge, so also is it with every soul that has a care to receive her proper food; wherefore when at last she has beheld Being she is well content, and contemplating truth she is nourished and prospers, until the heaven's revolution brings her back full circle. And while she is borne round she discerns justice, its very self, and likewise temperance, and knowledge, not the knowledge that is neighbor to Becoming and varies with the various objects to which we commonly ascribe being, but the veritable knowledge of Being that veritably is. And when she has contemplated likewise and feasted upon all else that has true being, she descends again within the heavens and comes back home. And having so come, her charioteer sets his steeds at their manger, and puts ambrosia before them and draught of nectar to drink withal.*

Plato makes a revolutionary statement here about the nature of scientific knowledge. He declares that one cannot know the physical world directly because it is, in fact, *less real* than the *hyperouranos*. As he puts it in *Timaeus*, a Form is "what always is and never becomes," and a particular is "what becomes and never is." Further, physical reality is merely the "receptacle of all Becoming," a three-dimensional *space* that affords spatial *extension* and *location* to all particulars. Unlike Forms, particulars are subject to *change*, and thus cannot be sources of absolute knowledge. They are slippery, metaphysically undefinable, and explainable only in terms of the unchanging Forms of which they partake.

The above passage can be interpreted as an early argument for basing human knowledge in abstract mathematics rather than in subjective evaluations of concrete observations. For example, to *know* what a triangle is, one cannot simply point to a triangular rock or even to a very carefully drawn diagram of a triangle. One must instead use *abstract reasoning* to express the Form of a triangle by means of a *formal definition*: a three-sided shape in a two-dimensional plane. This is the *essence* of a triangle. In contrast, the triangular rock and diagram each partake of multiple Forms, and neither are perfectly triangular. We may still *call* them triangles, but only those who know the Form of a triangle will understand what we mean when we do so. Therefore, it is knowledge of the unambiguous Form that gives insight into the complex and ever-changing particular, not the other way around.

Here, our object-property distinction blurs. Because while Forms may define universal properties, are they not also objects in an abstract sense? And while concrete particulars may be objects, are they not also fully characterized by a set of properties? In Plato's metaphysics, the categories of interest are really the *concrete* and the *abstract*, that which we *perceive* and that which we *conceive*. Plato further claims that humans come from the abstract realm, where they familiarize themselves with the Forms, and they are born into the concrete realm, where particulars are abundant and Forms are only latent memories. It is through philosophy then, the exploration of the rational mind, that we unearth our innate understanding.

To explain the link between the realms, Plato also postulates a *demiurge*, a transcendent artisan who shapes an initially chaotic space into an ordered, material reality, using the Forms as models. This entity is benevolent and wishes to craft a world like the ideal one he sees, but the objects that he *instantiates* are always imperfect copies. Like that of a master portraitist, his work is beautiful, but it is, in the end, a mere *image* of his subject.

So it is for programmers. Before us lies the space of computer memory, uniform and featureless, a receptacle of Becoming. Our Forms are *types*, our particulars are *values*, and the behavior in the space is dictated by the *code* that we execute. In *object-oriented programming* (OOP), values can be of an *object type* that is formally defined by a *class*.

For example, my cat Mystic belongs to the class of entities known as Cat, the set of all possible cats. She is an *object* known as a cat, and her catness is defined by the class Cat. In Plato's terms, she is a *particular*, partaking of the Form *Cat*. Just as the demiurge instantiates particulars from Forms, so the programmer instantiates objects from classes. Both bridge the gap between what is possible and what is.

That said, while Plato's metaphysics gives good insight into OOP and other abstract (or *high-level*) programming concepts, it cannot explain concrete (or *low-level*) ideas like *data* that is represented with *bits*. This is because Plato was not overly concerned with the *substance* of objects, the stuff that makes them *physical*. Thus, in our pursuit of a holistic understanding of *information*, we must also consider *substance theories*, philosophies that give metaphysical weight to *matter*.

1.2.4 *Aristotle's Hylomorphism*

Blah

1.3 Theories and Models

Blah

1.3.1 *Euclid's Elements*

Although arithmetic has been practiced since prehistory, theoretical mathematics was founded by the Ancient Greeks. The Greeks, in contrast to earlier peoples, applied *deductive reasoning* to the study of mathematics. While sophisticated arithmetic, geometry, and algebra was done before this period by the Babylonians and the Egyptians, this work was all based on *evidence* collected from the physical world. The Greeks were the first to recognize the need for *proof* of mathematical statements using *logic*.

While various proofs were written by Greeks in the centuries before his birth, Euclid of Alexandria is nonetheless credited with formalizing the concept of proof due to his use of the *axiomatic method* in his groundbreaking treatise, *Elements*. This work was the first of its kind, serving as both a comprehensive compilation of Greek mathematical knowledge and as an introductory text suitable for students. Additionally, it was a product of a synthesis of Greek thought, founded on the epistemology and metaphysics of Plato and Aristotle.

Elements is divided into thirteen books. Books I–IV and VI cover *plane geometry*, the study of shape, size, and position in two-dimensional space. Books V and VII–X cover elementary *number theory* (classically known as *arithmetic*) in a geometric way, expressing numbers as lengths of line segments. Finally, Books XI–XIII cover *solid geometry* by applying principles of plane geometry to three-dimensional space. Each book begins with a list of *definitions* (if necessary) that assign names to abstract mathematical ideas that are expressed in unambiguous terms. These ideas are then used to prove *propositions*, mathematical statements that warrant proof.

Euclid's proofs in *Elements* can be described as *axiomatic*. That is, they start from *axioms*, statements that are taken as true without proof. Axioms solve a problem in logical argumentation known as *infinite regress*, which often occurs, for example, when children ask questions. A curious child, seeking knowledge, might ask his parent why something is the way it is. That is, he requests a proof of some proposition P_0. The parent would respond with some proposition P_1 that supports the truth of P_0. The child, unsatisfied, would subsequently ask why P_1 is true. The parent would respond with the assertion that P_2 is true and implies P_1, and the child would similarly question whence P_2 arises. This dialogue could continue in this manner forever with the child endlessly searching for an anchor with which he can ground his understanding. However, patience waning, the parent must end this line of questioning by giving an unquestionable answer. Common examples include "Because that's just the way it is" or "Because I said so."

Some Greek philosophers considered infinite regress to be a serious epistemological issue. If we assume that all knowledge must be demonstrable by a proof or logical argument, we cannot know anything because our axioms cannot be satisfactorily proven. And yet, there are things that we claim to know. In his *Posterior Analytics*, Aristotle challenges this notion that all knowledge must be provable, averring that

> *All instruction given or received by way of argument proceeds from pre-existent knowledge.*

Aristotle continues his text in support of this statement. He discusses the nature of *premises* in his *syllogistic logic* and how they differ from the *conclusions* that they derive.

Namely, he states that the premises of a knowledge-producing deductive argument or *syllogism* must be *true*, to ensure the truth of their conclusion, *indemonstrable*, to ensure that they are independent of proof, and *causal*, to ensure that their conclusion logically follows:

> *Assuming then that my thesis as to the nature of scientific knowledge is correct, the premises of demonstrated knowledge must be true, primary, immediate, better known than and prior to the conclusion, which is further related to them as effect to cause. Unless these conditions are satisfied, the basic truths will not be 'appropriate' to the conclusion. Syllogism there may indeed be without these conditions, but such syllogism, not being productive of scientific knowledge, will not be demonstration. The premises must be true: for that which is non-existent cannot be known—we cannot know, e.g. that the diagonal of a square is commensurate with its side. The premises must be primary and indemonstrable; otherwise they will require demonstration in order to be known, since to have knowledge, if it be not accidental knowledge, of things which are demonstrable, means precisely to have a demonstration of them. The premises must be causes, since we posess scientific knowledge of a thing only when we know its cause; prior, in order to be causes; antecedently known, this antecedent knowledge being not our mere understanding of the meaning, but knowledge of the fact as well.*

Thus, Aristotle differentiates *pre-existent knowledge* from *scientific knowledge*. Whereas the latter is produced by means of a logical proof, the former is produced by one's *intuition*. He goes on to argue that what we intuitively understand is indeed knowledge, in the same way that what we prove is knowledge. The difference, he argues, is that axioms are *self-evident*. For example, the reflexive property of the natural numbers (i.e. for every natural x, $x = x$) is typically considered so obvious that proof is unnecessary. At the same time, the statement is so fundamental that it cannot really be proven. Despite their unprovability, Aristotle argues that axioms such as these constitute knowledge that is simply, unquestionably *known*:

> *Our own doctrine is that not all knowledge is demonstrative: on the contrary, knowledge of the immediate premises is independent of demonstration. (The necessity of this is obvious; for since we must know the prior premises from which the demonstration is drawn, and since the regress must end in immediate truths, those truths must be indemonstrable.) Such, then, is our doctrine, and in addition we maintain that besides scientific knowledge there is its originative source which enables us to recognize the definitions.*

Euclid apparently agreed with this epistemology because Book I of *Elements*, in addition to its definitions and propositions, contains a number of axioms, which are labeled either as *postulates* or *common notions*. In antiquity, a postulate denoted an axiom of a particular science, whereas a common notion denoted an axiom common to all sciences. Today, we do not think of axioms in this way. Rather, we simply consider them starting points for proofs. Euclid's axioms are given below:

Postulates

1. Hi

Common Notions

1. Hi

Euclid's proofs can also be described as *constructive*.

1.4 Modeling Physical Reality

1.4.1 *Pre-scientific Thought*

1.4.2 *Point Particles and Forces*

1.4.3 *Waves and Fields*

1.4.4 *Quantum States and Operators*

1.5 Information Theory

1.6 Signals and Systems

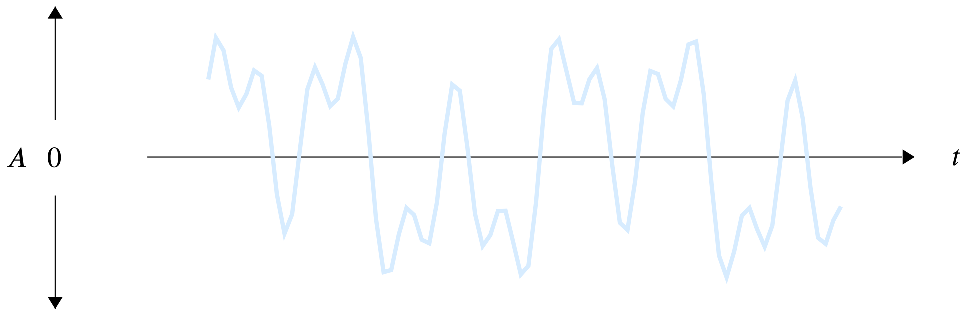
Data are the building blocks of *signals*, which are mathematical models of quantities that change over time. Signals are carriers of *information* about these quantities. A piece of information, known as a *message*, is encoded as data of a certain *medium*, either physical or numerical. In this sense, one can consider information a meaningful pattern that is embedded in a phenomenon. Philosophically, information is the *essence* of a *substance*, a set of properties that defines what a thing is. Mathematically, signals are functions f that map a set of times t to a set of amplitudes A, and they are classified according to the properties of these sets.

Sample rate is a property that is related to time. *Continuous* signals are functions of continuous time. That is, t is either the real number line $\mathbb{R}$ or an interval of it. In this case, t is also called a *linear continuum*. This definition of continuity is different from those given in real analysis, which describe functions for which sufficiently small changes in input result in arbitrarily small changes in output. For this reason, these signals are also called *continuous-time* in order to avoid any confusion with the continuous functions that are described in mathematics. If a signal is not continuous, it is *discrete* or discrete-time.

Continuous signals that have an infinite number of possible amplitude values (i.e. those that are modeled by $\mathbb{R}$ or $\mathbb{C}$) are called *analog* signals because an infinite-valued output can be viewed as *analogous* to the seemingly infinite complexity of real-world phenomena. At any instant in its time domain, an analog signal can *exactly* model a time-varying, real-world quantity, such as voltage, pressure, distance, mass, color, or anything else that is quantifiable. All continuous signals are analog, but not all analog signals are continuous. For example, the analog data of a continuous signal can be sampled at a finite rate in order to construct a discrete, analog signal.

In general, signals can be viewed as messages encoded in *data* whose decoded content is *information*. Like traditional written messages, ... Any signal can be expressed as an ordered sequence of data, each of which can be expressed as a tuple $(t, A(t))$. Because there are an infinite number of "instants" in any given interval of continuous time, the continuous signal shown above can be expressed as an *infinite*, ordered sequence of analog data.

Consider an arbitrary continuous signal:



Analog data are special because they exist in continuous space and thus theoretically have an infinite degree of precision. If we assume a classical model of physics in which time flows continuously, we can model physical quantities with real numbers by writing an equation $A = f(t)$ where $t, A \in \mathbb{R}$. In this sense, a complete set of analog data over a given time interval (i.e. an analog signal) can perfectly describe a physical quantity over the same interval. However, handling analog data can be very difficult.

Imagine that you throw a ball in the air and somehow exactly measure its height above the ground at every instant before it lands. If you were to organize these height data with respect to time, you would construct an analog signal that perfectly describes the height of the ball at any point during the throw. In practice, however, no measuring instrument has the required speed or precision necessary to do this. Additionally, you would need an infinite amount of memory to store continuous-time data that is collected at every instant.

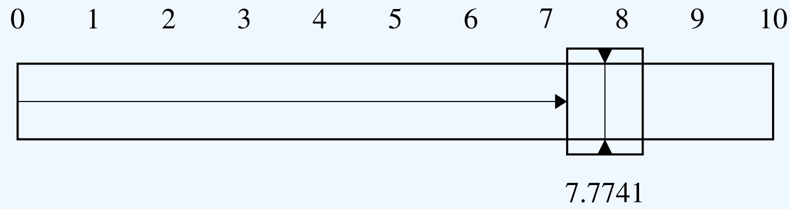
The Rationale for Binary Numbers

Binary encoding of data is not a requirement for computation. Rather, it is a concession that is made toward digital electronics. Computers have been designed for other bases, and, in some cases, a non-binary encoding is actually preferable. As long as the data can be encoded with a finite number of symbols, it can be handled by a *digital* computer (i.e. a computer that manipulates digits). A digital computer receives and operates on a *discrete* signal of data, a value or set of values that updates at regular time intervals (e.g. every second, every minute, etc.).

One could argue that *analog* computers process data that are encoded with an infinite number of symbols. However, this would require a loose definition of "symbol" because analog data cannot be perfectly represented by digits. Rather, analog data is represented in a real, physical medium, like position, pressure, or voltage. An analog computer receives and operates on this *continuous* signal of data that updates instantaneously with real-world time.

Consider the Mark 1A Fire Control Computer, a device, more appropriately termed a "calculator" by modern definitions, that was installed on United States battleships in World War II. This machine used mechanical parts, such as gears, cams, followers, differentials, and integrators, to perform the arithmetic and calculus necessary to hit an enemy ship with a projectile, even if both ships were moving in different directions. It also used smooth components that gradually translated rotational motion into linear motion along a continuum.

Despite being labeled with decimal (base-10) integers, this tool actually received *continuous*, *analog* quantities, such as physical position, as input. If each of its mechanical positions were considered a different "symbol," this system would theoretically be able to represent uniquely every real number to a infinite degree of precision and, thus, compute continuous functions without error. Such analog computation is performed also in slide rules, as well as in early electronic computers, which used voltage to represent numbers instead of position.



Any digital encoding of numbers is possible in computing, and the choice does not affect what a computer theoretically can or cannot do. Rather, it has practical consequences. Decimal encoding was implemented in a number of early electronic and electro-mechanical computers, such as the Harvard Mark I and the ENIAC.

Regardless of which encoding is used, data is data.

Data are unorganized facts or statistical values that may reveal something of interest. They are collected by *sampling* the properties of *phenomena* from the observable Universe. Once collected, they can be cleaned, organized, and then analyzed in the hope of gaining a better understanding of reality. When a set of structured data is put into context, it can be interpreted as *information* that explains what has been observed. This information can then be *encoded* as an ordered sequence of data known as a *message*. Messages are *transmitted* and *received* both by Nature and by Man. In the latter case, this process can be viewed as the *communication* of abstract, human thought.

Data can describe a variety of things, such as quantities, qualities, and relations. In computer science, data are *quantitative*. That is, they describe *quantities*, which are properties that can be modeled with *numbers*. A quantity can be further classified as either a *multitude* or a *magnitude*. The former is a *discrete* quantity that is *counted*. One would ask "how many" elements belong to a multitude. In contrast, a magnitude is a *continuous* quantity that is *measured*. It is the size of something that is modeled as a *continuum* (e.g. by fields such as $\mathbb{R}$ or $\mathbb{C}$), and one would accordingly inquire as to "how much" of it there is.

A set of data collected at a finite rate (e.g. by a *measuring instrument*) is technically a multitude of samples, even if the quantity being measured is a magnitude. For example, if a ball were thrown in the air, one could ask "how much" height it has at regular intervals of time. Once the data collection ends, one can also ask "how many" samples were acquired.

Furthermore, one can ask "how often" samples were taken. This is an evaluation of the *sampling rate*, and it is really just an alternative way of asking with "how much" speed samples were taken. Rates and ratios are still quantities. The existence of the rational numbers $\mathbb{Q}$ are evidence enough of this.

Data can be represented by various *media*, both symbolic and physical. In the age of modern computing, data are often assumed to be *digital* in nature, expressed in the symbolic medium of numerical *digits*. This is the case for data processed by modern, electronic computers, which charge capacitors in order to generate voltages that represent *logic levels*. A *transistor* acts as a gate to each capacitor, staying closed to retain *electric charge* during storage and opening to measure or modify voltage on reads and writes respectively.

The ratio of a capacitor's charge to its capacitance determines its voltage (i.e. $V = \frac{q}{C}$). This voltage is then mapped to a logic level, which corresponds to a range of voltage values. These logic levels represent digits. In modern computer memory, data are represented as binary digits or *bits* (i.e. the digits 0 and 1), each of which represents a binary number. In this case, there are two logic levels, *low* and *high*.

Table 1: Voltage Ranges for CMOS Logic Levels (V)

Signals	Low (0)	High (1)
Input	0.00–1.50	3.50–5.00
Output	0.00–0.05	4.95–5.00

However, quantitative data need not be expressed with symbolic digits. They can instead be defined in terms of an *analogous* physical quantity. For example, the unit of measurement known as the *millimeter of mercury* (mmHg) defines pressure in terms of the height of a column of mercury in a manometer. It is used to quantify blood pressure, and it does so by considering height an *analog* or *model* of pressure measuring that instead. Using this method, data about one phenomenon can be encoded as data of a different phenomenon, provided that the quantities involved are somehow related to each other. In this case, a height of 1 mm is related to a pressure of 133.322 Pa above atmospheric pressure.

1.7 Semiotics, Language, and Code

However, the term "universal language" is also often used in other contexts to describe a hypothetical language that all of the world speaks and understands. This is an old idea, and it is addressed in a number of myths and religious texts. In Genesis, the myth of the Tower of Babel explains the diversity of language as a result of God thwarting the Babylonians' plan to build a tower that would extend into the heavens. As punishment for their blasphemy, God "confused their tongues" and dispersed them across the world, shattering their once universal language. Similar myths are told by cultures across the world, such as the Native American Tohono O'odham people, who tell tales of Montezuma attempting to do the same, attracting the ire of the Great Spirit.

Alas, no such language exists and it is unlikely that one has ever existed. Rather, human language is generally believed to have evolved independently around the world from prelinguistic systems of communication such as gesturing, touching, and vocalization about 100,000 years ago. Similarly, written language evolved from proto-writing, the repeated use of symbols to identify objects or events.

Writing is distinct from speaking in that it is a reliable method of storing and transmitting information. Before the invention of written language, important pieces of history and literature were preserved through *oral tradition*, passing from one generation to the next through repetition and memorization. However, oral tradition is prone to data loss and unintended changes. Like messages in a game of telephone spanning centuries, stories were at risk of losing old details and gaining extraneous ones. Additionally, if a society fell apart, their stories could be lost forever. Writing is a tool that mitigates these issues by *encoding* speech into a symbolic code that can be inscribed onto durable media, such as clay tablets or stone, or onto more delicate, portable media such as parchment or paper.

Symbology in Natural Languages

A *symbol* is a syntactic mark that is understood to represent a semantic idea. For example, the numeral "2" is a symbol for the abstract concept of the number known as "two," the quantity of items in a pair. Symbology is universal for mankind, as it is an exercise of our capacity for abstract thought. Humans can leverage the pattern-recognizing power of the brain to associate symbols or sequences of symbols (also known as *strings*) with ideas. Thus, we can proliferate information textually by means of a *writing system*.

Languages are diverse, and so are their writing systems.

An argument of any kind must be expressed in a *language*. In the case of an informal argument, such as a debate, a *natural language* is typically used (i.e. a language, spoken or written, that evolved *naturally* as an aspect of human culture). This is done in the name of *universality*. For example, a political debate that is held and perhaps broadcast in Poland would likely be conducted in Polish because Polish politics are primarily of interest to Polish people. Many Poles speak English in addition to Polish, but there is no language more widely spoken in Poland than Polish, with a whopping 97% of the country declaring it as their first language. For media directed toward the population of Poland, there is no language that will better ensure an effective dissemination of ideas. It is a nearly *universal* language in this context, a language understood by all intended recipients.

Formal proofs, on the other hand, require a *formal language* for clarity and precision. In practice, formal proofs are rarely written or read by logicians or mathematicians. Typically, mathematical proofs are written using a combination of formal and natural language.

However, computer programs, written in *programming languages*, are considered formal.

A formal language is constructed from an *alphabet* Σ, which is a set of symbols. The set of all possible finite-length *words* or *formulas* that can be built from this alphabet is denoted Σ^*. A formal language L over an alphabet Σ is then defined as a subset of Σ^*. Thus, a language is a set of purely syntactic objects. It may conform to a *grammar* that specifies how words can be produced or arranged in a *sentence*, but a language is never inherently meaningful. The semantics of a language is always interpreted separately from the syntax.

2 REASONING AND LOGIC

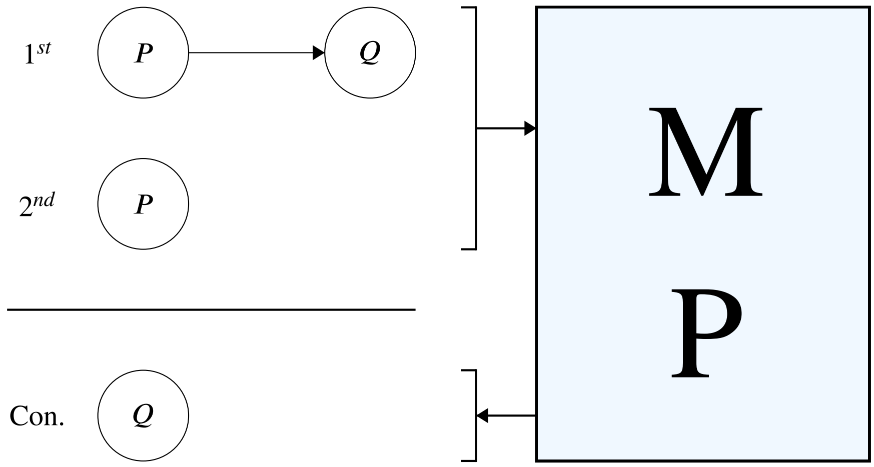
2.1 The Syntax and Semantics of Argument

Reasoning is a cognitive act performed by rational beings. It involves the absorption and synthesis of presently available information for the purpose of elucidating knowledge. It may require the ingenuity of thought, but in some cases the rote application of simple rules is sufficient. Reasoning could also be described as the providing of good *reasons* to explain or justify things. However, as you might expect, there is much debate over whether or not any particular reason is "good." Reasoning is broad, and it comes in different flavors, but ultimately it is the pursuit of *truth*.

Reasoning may involve the application of *logic*. This *logical reasoning* is the kind of reasoning that can be expressed in the form of an *argument*. However, this argument does not need to be "correct." Such reasoning simply must be explainable in *steps*, whether that be through informal speech or formal writing.

In logic, an argument has two parts: a set of two or more *premises* and a *conclusion*. If the conclusion *necessarily* follows from the premises, the argument is called *valid*. To know whether or not an argument is valid, one must produce a step-by-step *deduction* of the conclusion from its premises.

A deduction is constructed within a *deductive system*, which also has two parts: a set of two or more *premises* and a set of *rules of inference*. A rule of inference can be thought of as a kind of logical function. It takes statements (such as premises) as input and outputs either an intermediate result or the conclusion. A deduction, then, has *three* parts: a set of two or more *premises*, a *conclusion*, and a set of *intermediates*. A single-step deduction is represented below, along with a suitable rule of inference labeled MP. Note that the set of intermediates is empty in the case of deductions with only one step.



This is the quintessential rule of inference used in logical reasoning, expressed here in symbols. It is known as *modus ponens* (Latin for "a method that affirms"). The first premise $P \to Q$ is called a *conditional statement* where $\to$ is the *implies* operator. It states that if the *antecedent* P were true, it would *imply* that the *consequent* Q would also be true. The second premise P simply states that the statement P is indeed true. Thus, by means of *modus ponens*, we can *infer* from the premises $P \to Q$ and P that the conclusion Q must be true.

Inference is also called *entailment*. Though these terms describe the same logical concept, the latter more aptly describes the relationship between statements that are connected in a deduction. To infer is to come to a conclusion that is not *explicitly* stated in the premises. To entail is to require that something follows. If P *entails* Q, this suggests that Q is a *logical consequence* of P, which we can express using a *turnstile*: $P \vdash Q$. Inference is an action that humans perform. Entailment is a relationship between logical rules.

When the inferences made mirror the entailment, you've got yourself a *valid* argument. However, this does not necessarily mean that your argument is *sound*. *Validity* is a property of arguments with a conclusion that can be *deduced* from its premises according to rules of inference. However, validity says nothing about whether or not the premises have a *meaning* that accurately describes reality. *Soundness*, on the other hand, is a property of valid arguments whose premises are *known truths* or *axioms*. The deduction of a sound argument is called a *proof*.

Validity and soundness complicate entailment. Now, an argument can be evaluated either in terms of its *syntax* or in terms of its *semantics*. Syntax refers to the arrangement of symbolic *words* (or, more generally, *tokens*) in a body of text. Semantics, on the other hand, refers to the *meaning* that can be *interpreted* from the text. Both of these concepts are integral to human language and indeed to the communication of information in general.

Consider the following valid deduction:

"If it is raining, it is Tuesday."	$P \to Q$
"It is raining."	P
"Therefore, it is Tuesday."	$\therefore Q$

This conclusion is a *syntactic consequence* of the premises ($P \vdash Q$). That is, when the argument is evaluated simply as a collection of symbolic strings that are manipulated according to rules, P entails Q.

Syntactic consequence is sometimes expressed with the following statement: "If the premises are true, the conclusion must also be true." However, while this statement does hold for all valid arguments, text evaluated syntactically does not have any notion of truth. It is simply a sequence of tokens. Thus, it may be better to think of valid arguments as "games that are played according to the rules." For example, chess has no inherent meaning associated with it, but it still has rules, and valid games of chess are those that comply with those rules.

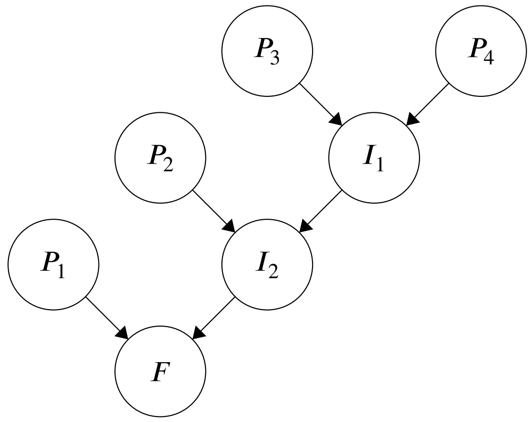
Now, it might be Tuesday and raining as you are reading this. However, semantically, it is not required to be Tuesday if it is raining. The first premise given above is, in reality, false. However, if we change it to something that we consider true, we can come to a conclusion that is both valid and sound:

"If it is raining, it is wet outside."	$P \to Q$
"It is raining."	P
"Therefore, it is wet outside."	$\therefore Q$

In this case, the conclusion is a *semantic consequence* of the premises ($P \vDash Q$). In addition to the argument being syntactically logical, it is also semantically true "in all universes." That is, there is no possible scenario where it is raining, and it is not wet outside. This syntactic-semantic difference will come up repeatedly throughout this guide.

2.2 The Structure of Proof

Of course, arguments can have more than one step. An argument can involve many inferences, some of which may use *intermediate conclusions* as premises for further conclusions. Multiple threads of reasoning can extend from the premises, weaving through intermediate conclusions and intertwining via inference until they all meet at a *final conclusion*. This is the *tree structure* that is inherent in a *formal proof*.



In this *tree*, the *first premises* are labeled as P_n where $n \in \{1,2,3,4\}$, the *intermediate conclusions* are labeled as I_m where $m \in \{1,2\}$, and the *final conclusion* is labeled F. Though one can make a distinction between first premises and intermediate conclusions, it is important to note that they are all premises with regard to the eventual final conclusion. Thus, there are no restrictions on combining them beyond those dictated by the rules of inference.

In modern proof theory, proofs like these are studied not only for their semantic meaning, but also for their mathematical structure. They are treated like *data structures*, which are abstract objects that are used in computer science to model the storage and flow of digital *information*. Similarly, a proof is an abstract text that models the flow of logical information as it is manipulated by rules of inference toward a final conclusion.

Information, in the context of computer science and, more generally, *information theory*, refers to a *syntactic* message. Formally, it is an order sequence of symbols

In computer science, the data structures of interest are *recursive*. That is, the structure can be defined "in terms of itself." *Recursion* is a deep topic, and we will spend much of the next part characterizing it, but a practical way to think about a recursive structure is in terms of *type*.

Types are a widespread feature in programming languages. As a concept, they are founded in a logical system called a *type theory*. There are multiple type theories, and some of them are used as alternative foundations for mathematics. We will cover all of this in detail later, but, essentially, a type is a label that can be given to

However, not all forms of reasoning have such strict rules. There are other methods of reasoning that cannot be modeled by an argument. In contrast to logical reasoning, *intuitive reasoning* has steps that are *not* understood. Although the question might seem peculiar, it is worth asking whether or not computation can be intuitive. So, to begin our journey of understanding computation in a modern, logical sense, we will first walk in the other direction, considering it instead in a mystical, otherworldly sense.

2.3 The Oracle, the Seer, and the Sage

Intuition is the capacity to create conclusions without evidence, proof, or a combination of the two. If any premises are involved, they may appear to an onlooker as if they were plucked out of thin air. If any method is involved, it is esoteric or hidden from sight. Acts of intuition range from the mundane procedures we perform without thinking to great feats of intellectual, artistic, and athletic achievement. And while it is often associated with magic or supernatural ability, intuition is a real, observable phenomenon, and it is a form of reasoning.

Like that of reasoning, the definition of intuition is fuzzy. There are a variety of events that one might label as the product of intuition that are actually quite functionally distinct from one another. For this reason, I would like to consider and compare three archetypes that are known for their intuitive skills: the Oracle, the Seer, and the Sage. For the skeptics among you, I ask that you suspend your disbelief for a moment and assume that our characters are acting in good faith. There is no lying here; the conclusions are sincere.

THE ORACLE

An Oracle is a person who predicts the future by acting as a vessel for a god or a set of gods. They are considered by believers to be portals through which the divine speaks directly. They are found in histories all over the world, but most people associate the role with the priestesses of Ancient Greece. Picture a woman with eyes that roll back into her head as she speaks in a possessed, thundering voice.

For an Oracle, the conclusions come straight from the source. She may not even do any reasoning herself, save for the *unconscious reasoning* that is performed while she is possessed. However, conscious or not, she provides a great service to mankind. There is no clearer answer than the one given to you directly from the gods themselves, even if it has to be sent through what is essentially the human version of a telephone.

For those seeking counsel, the premises of the Oracle's conclusion are unnecessary because they just *know* that the statement must be true. For them, the connection the Oracle has with the gods and with reality is part of life itself. We all have deep beliefs like this. For example, most people do not feel that gravity needs to be proven to them in order for them to accept it as fact. Their perception of gravity in the world around them is proof enough. This is the kind of intuition characterized by *subconscious reasoning*, the reasoning you do without being aware of doing it.

THE SEER

A Seer is a person who predicts the future by interpreting signs from a god or a set of gods. Unlike an Oracle, a Seer speaks divine truth in his own words, drawing from an innate, sometimes god-given power to see meaning in natural events or occult objects. There are various methods of divination that are used by Seers, such as scrying (the gazing into magical things, like crystal balls, for the purpose of seeing visions), auspicy (the interpretation of bird migration), or dowsing (the use of magic to find water, often with the aid of a dowsing rod). The acceptance of any particular method is cultural, but beliefs in divination vary greatly, even within a single society.

The Seer has premises for his conclusion, but the rules of inference involved in his reasoning are incomprehensible to others. He

THE SAGE

What is Truth?

Truth, in the absolute sense, has been discussed and debated since the dawn of man. It is a concept that seems obvious to us, and yet it always seems to elude our understanding. Philosophers have formulated dozens of theories of truth over the years, with some asserting that truth is an objective property of our universe and others asserting that truth is a useful lie that mankind has invented. And while there is merit in those claims that truth is not real, we still see around us the technology that was born out of our intuitive sense of true and false. Especially in computer science where everything is represented in binary.

The traditional theories of truth have been termed *substantive*. That is, they assert that truth has some basis in reality and that it is a meaningful thing to discuss. Early theories of truth from Ancient Greece are considered the foundation of *correspondence theory*, the idea that statements are true if their symbols are arranged in such a way that they express an abstract thought that accurately describes reality. This theory defines truth in the context of a relationship between language and objective reality. It is a useful philosophy that allows us to orient ourselves in the world around us. However, many philosophers believe that truth cannot be explained by such a simple rule. In fact, there are many more factors that could play a role in the concept.

Objections to a strict correspondence theory usually take issue with the treatment of language as something monolithic and easy to classify within a true-false dichotomy. For example, a statement encoded as a sentence in a language is only meaningful to people who can read that language. What does this imply about a particular language's relationship with truth? Is a statement only true for those who understand it? Furthermore, there is no guarantee that readers of the same language will even be able to agree on the meaning expressed by a particular sentence. Languages often encompass multiple dialects that might parse words differently. Words themselves also may not be precise, and some abstract thoughts may not have suitable words in some languages. These are the points raised in *social constructivism*, a theory which avers that human knowledge is historically and culturally specific. Social constructivists also believe that truth is *constructed* and that language cannot faithfully represent any kind of objective reality.

Other substantive theories find the essence of truth nestled in other abstract concepts. *Consensus theory*, as the name implies, defines truth as something that is agreed upon either by all human beings or by a subset (such as academic groups or standards bodies). This is another anthropocentric definition that is at constant risk of philosophical division on any given topic. *Coherence theory* takes a more objective approach, claiming that a statement can only be true if it fits into a system of statements that support each other's truth. This is similar to the notion of *formal systems*, which we will discuss in depth later. However, traditional coherence theories attempt to explain all of reality within a single coherent system of truths, which is incompatible with our modern understanding of formal systems.

Modern developments in philosophy have resulted in theories that deviate from the long-held, substantive opinions on the nature of truth. These *minimalist* theories assert instead that truth is either not real or not a necessary property of language or logic. They claim that statements are *assertions* and thus are implicitly understood to be true. For example, it is understood that by putting forth the sentence "$2+2=4$," you are endorsing the semantic meaning "$2+2=4$ is true." The clause "is true" is called a *truth predicate*, and minimalist theories of truth often consider its use redundant.

This idea of a truth predicate is borrowed from Alfred Tarski's *semantic theory of truth*, which is a substantive theory that refines the correspondence concepts espoused by Socrates, Plato, and Aristotle for formal languages. This theory makes a distinction between a statement made in a formal language and a truth predicate, which evaluates the truth of the statement. Tarski made this distinction in order to circumvent the *liar's paradox*, which is often presented with the following example: "This sentence is false." If the predicate "is false" is considered to be part of "this sentence," the truth of this statement cannot be decided. For this reason, Tarski states that a language cannot contain its own truth predicate. This is enforced by requiring that "this sentence" be written in an *object language* and that "is false" be written in a *metalanguage*. *Convention T*, the general form of this rule, can be expressed as

"P" is true if and only if P

where "P" is simply the metalanguage sentence P rendered in the object language. That is, the *syntactic representation* is assigned a *truth value* of true if and only if the semantic meaning it represents is considered true (according to whatever theory of truth you employ). By this rule, we say that "$2+2=4$" is true if and only if *the sum of the number* 2 *with the number* 2 *is equal to the number* 4. Minimalist redundancy theories modify Tarski's theory by interpreting "P" as an implicit assertion of the truth of P. The sentence "P," which asserts that P is true, is then false if and only if its semantic meaning P is false.

The debate on the nature of truth rages on *in perpetuum*. However, for our purposes, we must be practical. There is truth in logic and mathematics and computer science as well, but, in practice, it has little to do with the debates on *absolute truth* described above. Their truth is *relative*. That is, it relies on the assumption that our premises are true. Cognizant of this, we move forward with our thinking, searching for truths within these arbitrary boundaries.

2.4 The Mathematician and the Scientist

Logical reasoning is the *inference* of new information from present information. It involves *rule of inferences* that are used to relate sets of *premises* to *conclusions*. There are three kinds of logical reasoning: *deductive*, *inductive*, and *abductive*. Each is classified according to which piece of information (premises, rule, or conclusion) is missing and must be inferred from the others.

Inductive arguments exist on a spectrum between *weak* and *strong*. Those that are stronger and more persuasive have a higher probability of having a true conclusion. Inductive reasoning is associated with *science* and *critical thinking* because it allows one to make generalizations about complex phenomena given limited evidence. Unlike deductive reasoning, it attempts to find new knowledge that is not simply contained within its premises.

Statements made by induction are bolstered with evidence whereas deductive statements are as true as their premises. This leaves inductive reasoning susceptible to *faulty generalizations* and *biased sampling*. Induction must also assume that future events will occur exactly as they have in the past, which is not always the case. For example, a turkey that is fed every morning with the ring of a bell may infer by induction that bell → food. However, he will see the error in his reasoning when the farmer rings the bell on Thanksgiving Day and instead slits his throat.

ABDUCTIVE REASONING

Abductive reasoning is the inference of a *premise*, given a conclusion and a rule. It is investigative in nature. For example, given a conclusion "The grass is wet" and a rule "When it rains, the grass gets wet," we might determine that rain is the best explanation for the wetness of the grass. Thus, we abduce that "It might have rained."

Like induction, abduction can also produce a hypothesis. However, abduction does not seek a new relationship between two previously unconnected statements. Rather, it uses established relationships to find a reasonable explanation for a statement that is assumed to be true. It is often used by detectives or diagnosticians who need to find a probable cause of an event. It is also used in Bayesian statistics. While multiple premises may be abduced, typically we want to abduce a single, "best" premise.

Abductive reasoning allows us to ignore the many causes that are unlikely in favor of those few that may be relevant to the problem at hand. For example, doctors are often taught to heed the following proverb: "When you hear hoofbeats, think of horses, not zebras." That is, when a patient exhibits certain symptoms, a doctor should abduce from them a commonplace disease before considering more exotic possibilities. However, "zebras" do exist. Sometimes the most likely cause is not the actual cause. For this reason, abduction is also considered to be equivalent to a deductive fallacy called *affirming the consequent*, which is like a *modus ponens* performed in reverse. That is, given a conditional $P \rightarrow Q$ and the consequent Q, abduction infers P from Q by assuming that the converse $Q \rightarrow P$ of the conditional is also true. This is not a deductively valid inference. Considering our example again, the grass might be wet from rain, but this is not *necessarily* true. It is also possible that the sprinkler system is on. Or perhaps there is a zebra-esque scenario like a flood.

Induction and Abduction

TEXT

2.5 Mechanical Computation

Before we discuss and classify automata in depth, we should first consider what is **not** an automaton. What is an example of something that might perform some kind of calculation, but is not a computer? What about a microwave? Is a microwave a computer? No, it is not. A computer can be programmed in some meaningful, robust way. A microwave contains a microprocessor, which uses *combinational logic* and basic binary inputs to set timers and operate the oven. It cannot be programmed in any meaningful way. What about a calculator? Is a calculator a computer? If we are being formal, the answer is no, but it depends on what kind of "calculator" we are talking about.

Calculators, such as *counting boards* and the *abacus*, have been around since prehistory. Calculators with four-function capabilities have been around since the invention of Wilhelm Schickard's mechanical calculator in 1642. In the late 19th century, the *arithmometer* and comptometers, two kinds of digital, mechanical calculator, were being used in offices. The Dalton Adding Machine introduced buttons to the calculator in 1902, and pocket mechanical calculators became widespread in 1948 with the *Curta*. None of these are computers.

The difference between a calculator and a computer is that a computer can be programmed and a calculator cannot. What does it mean to be programmable? That is perhaps the central question of automata theory, and we will discuss in this section several levels of "programmability." However, for now, we can certainly say that a simple, four-function electronic calculator is not a computer. It simply uses combinational circuits like full-adders, full-subtractors, multipliers, and dividers to implement its functions, and there is no potential for modifications or user-defined functions.

Surprisingly enough, *special-purpose computers* have also been around for a long time. Early examples include the *Antikythera mechanism* (an Ancient Greek analog computer), Al-Jazari's 1206 water-powered *astronomical clock*, the 1774 Writer Automaton (a mechanical doll that could be programmed to write messages), the 1801 *Jacquard loom*, and *slide rules*. Some later mechanical computers were quite powerful, such as *differential analyzers* (special-purpose computers for solving differential equations) and fire control computers. Charles Babbage designed the *Analytical Engine*, a general-purpose computer, in 1837, and Ada Lovelace wrote a program for it, but it was never built. General-purpose computing would not reemerge until the 1940s.

The line between calculator and computer began to blur with the introduction of *programmable calculators* in the 1960s. Many modern high-end calculators are programmable in Turing-complete languages such as TI-BASIC or even C or C++, which officially makes them computers. Once we start implementing *sequential logic* with components like SR latches or D flip-flops, we are storing state, and state is a requirement for computing. Circuits that use sequential logic can be considered automata and, given enough complexity, computers.

Modern Computing is American

TEXT