1. The Matter of Computation

The magic is that, as it happens with cars, most people can use computers without understanding them.
Humans and animals alike are all remarkable users of information, yet no one can define exactly what it is.

Empirical facts

Unlike the emphasis of previous generations in science and engineering, the fundamental societal concern of our time no longer seems to be matter-energy, machinery or electronic hardware, or even biological function or bioware. The products of scientific and technological advances that brought about the digital age have already penetrated homes and have become an integral part of human lives. The impact was all but unpredictable less than two decades ago. As a consequence, a new world is emerging before our eyes. Distances have been eliminated by new entities such as the Internet, the Web, smart phones, and instant communication around the globe. The computer age marks a significant qualitative change in the old subject matter of "calculating machines" being run at human pace that inspired early computing devices centuries ago and precise definitions of computing nearly a century ago. This new world is sometimes referred to as cyberspace. It makes one wonder, what exactly has our subject matter become and what does it have in store for us in the near future?

A quick run through history may give some insight. At the end of the 19th century attention was focused on electromagnetic forces. The first part of the 20th century saw the emergence of energy, its understanding, and production. The later part of the 20th century saw the emergence of digital devices, their applications in "artificial intelligence" products such as chess player `DeepBlue` and Jeopardy contestant `Watson`, both capable of defeating human players, and their implications in terms of the human role in the universe. Each of them, energy or electronics, gave impetus to a branch of science that made possible its understanding. Analogously, the single fundamental commodity of our time in cyberspace appears to be *information*. Information is but universally transacted and accepted by society at large, despite the fact that matter and energy still remain major but secondary factors controllable through information processing.

The purpose of this book is to initiate the reader into the science of computing. The purpose of this chapter is to provide appropriate background for a definition of computer *science*, as the most appropriate context for the issues addressed in this book.

1.1 What exactly is information?

The obvious definition of computer science would would appear to go along the lines of *the science concerned with the study of computers*. This definition turns out to be inadequate for a variety of reasons. To begin with, we do not have car science, or microscope science, or even human science. But we do have computer science. More importantly, the concept of *computer* would have to be defined properly. The easiest way would appear to be, it is an *information processor*, which leads us right back to the grand question in the title of this section. If we could answer it, the ideal definition of computer science would then be along the lines of *the science concerned with the study of information and its processing.*

It is thus not a surprise that there have been various ways to approach a rigorous study of information. One is to attempt to develop some sort of theory of information, akin to what Euclidean geometry does for basic geometric intuitions concerning the phenomenon of areas and volumes, so important to tax assessors in ancient Egypt, using the kind of fictions (lines, areas, and volumes) encountered now in high school. Much later, Claude Shannon developed a theory in the 1940s that has become known as "information theory." However, it has been enormously useful more as a theory of data communication than a theory of information. Bateson's definition of information as "the difference that makes a difference" assumes again an observer to whom the difference is made, in other words a kind of information processor in itself (Bateson & DOnaldson, 1972). The definitions are somewhat circular and not so scientifically sound. More recent attempts at developing an appropriate theory of information expose deeper difficulties. Chapter 1 in (Devlin, 1991), provides a sobering assessment of the magnitude of the challenge via the metaphor that we would understand today as much about the answer to "What is information?" as an Iron Age man understood in his time about the answer to "What is iron?". The best answer that we know requires, at least, knowledge of the atomic structure of matter. Other authors have further explored similar arguments with similar or more radical conclusions, ranging from a general theory of information (see e.g., Burgin et al, 1999, 2012, 2011), to sheer impossibility of a unified theory of information (Capurro et al, 1999). Therefore, the definition of the subject matter of computer science as information and its creation and transformation would seem to lead to a deep dead end for the foreseeable future. And yet, we can sense the phenomenon of "information" lying there in the world, staring at us from every living brain in the guise of data, knowledge and wisdom, and manipulated in an enormous variety of complex forms and ways. So like the ancient Greeks, we will need to continue to use and study "information" in the intuitive sense afforded by our experience of it, not by the theories that we may be able to articulate about it.

Therefore, it is not surprising that the electronic age has been born more directly concerned with information processing devices, particularly in their specific electronic implementation, i.e., the conventional computers of our time. A major theme is thus the design and construction of electronic data processing machines. Such devices are called automata, computers, and more recently, smart phones and agents, especially when they are confined to a particular domain of application. The branch of science that has developed around this effort is primarily known as *computer science*. The ubiquity, importance, and uniqueness of the new tool are reflected in its singular name. We do not have car science, or microscope science, or even mammal science. But we do have computer science. Why could that really be?

Upon reflection again, this is not surprising in view of the characterization above. If the subject matter of computer science is information, then computer science must be a second or higher order science because every natural science transacts in information of a particular type: *physics* about time, space, and matter/energy in general; *chemistry* about the composition and transformation of matter; and *biology* about organic matter in the form of what is called life. For example, cyberspace now includes most scientific content today, regardless of the science, and we expect a search engine to return answers about scientific questions as well. Therefore, an appropriate characterization of current computer science cannot focus on specific types of matter (e.g., silicon, or biomolecules, or subatomic particles), although that type of focused research can certainly enlighten the confines and boundaries of the field. Later developments confirmed that the pioneers in the field of classical computing (that largely defines it today, e.g. as in Hopcroft and Ullman [8]) rightly recognized that the abstractions of mathematics and logic were the appropriate setting to develop a science of computing. Since information encompasses concepts even more abstract than taxes, energy, and even quantity and number, it would appear necessary to go even higher in abstraction for a theory of information. On the other hand, as hinted by the dynamics of cyberspace, the past models and theories in classical computing are falling shorter and shorter of encompassing most of what is considered computing today. And as Dijkstra pointed out early on, their relation to the science of computing should not be any tighter than that of telescopes to astronomy. Which way to go in moving forward?

There is a variety of reasons for a science of computers. The most evident is the versatility and universal applicability of computers. Like energy, information is stored, transmitted, transformed, and manifested in many ways, if properly anchored in physical reality. Unlike energy, it seems information can be created, copied, hidden, and destroyed. Information is more abstract than energy, and it certainly can be used to control and manipulate energy and ordinary physical objects. That is the ultimate reason why information processing machines, computers in particular, have a versatility and dexterity unprecedented in history. They allow simulation of virtually everything, from physical phenomena such as wind tunnels and weather, through genetic interactions, through chess players and, according to some, even full human brains. This influence shows no indication of abating and, in fact, appears to have the potential to engulf society. It is thus perhaps inevitable that we will find an increasing demand for a solid science that provides us with understanding and inspiration to dream up better and more powerful computers today that may perhaps be produced for availability a few months or decades from now. In this Section, we will explore what appears to be the most appropriate answer if the overarching eventual goal of computer science is to elucidate the nature of information, including information processing devices (computers), not only as they exist in seller's catalogs, but more importantly, in their fundamental nature as a scientific and technological quest.

1.2 What should be in a Model?

How then to undertake a study of this nature? Historically, beginning with early models in physics, understanding any type of phenomenon, be it physical, chemical, or biological, has amounted to the construction of *models* of the phenomenon that

- can play an explanatory role (what's going on);
- have relatively simple components (how to build them?);
- afford a predictive power and a useful role (how to use them to enhance our lives?).

For example, the phenomenon of areas and volumes so important to tax assessors in ancient Egypt, is much better understood through the kind of fictions encountered in euclidean geometry. The phenomenon of energy is understood thorugh the study of idealized gases in thermodynamics, and currents and voltages in electromagnetism. Models are themselves idealized simplifications of the enormously complex external reality for the purpose of making it accessible to the human mind. The 'points' and 'lines' that euclidean geometry talks about may not exist in reality outside human minds. No object falls according to the equations of free fall either on earth or far in space, but Newtonian mechanics gives an amazingly close enough approximation most of the time. An understanding of information should do likewise. Bateson's definition of information as "the difference that makes a difference" assumes an observer to whom the difference is made, in other words a kind of information processor. It is therefore a somewhat circular and not a quite satisfactory definition. And since information perhaps encompasses concepts even more abstract than taxes, energy, and even quantity and number, it is likewise necessary to create models of information processing machines definable from more primitive concepts in order to understand the nature of information and its processing.

These models have remained appropriate even though they require a human brain to be used, as noted above. Furthermore, increasing use of automation seems to be pointing to the need to build some type of intelligent control or "brain" in these models. Therefore, creating models of information processing is an incredibly complex task in its full scope because even human brains and minds may be construed as information processing devices. If so, trying to create a brain is somewhat analogous to having a personal computer trying to create a better computer than itself by itself. So, for the first time in history, we are no longer trying to understand objects and processes outside the human mind, but something awfully close to the human brain/mind itself. Real human brains/minds might not be appropriately regarded as such models, since they may be too complex to permit the type of manipulation and understanding that models should provide. We thus find ourselves facing another forbidding task, re-inventing brains, a task that is almost as daunting as finding a sound definition of information.

A good model has to keep a balanced tension between its complexity and understandability, on the one hand, versus the complexity of the real phenomena it should explain and predict, on the other. A computing device must therefore include at least three basic components:

- Some mechanism for reading and *encoding* (one might call it *perceiving* input information from the external world, which in turn requires things like keyboards, sensors, memory, activation levels, internal states, and other computational primitives;

- Some *processing* mechanisms that dynamically changes these internal states, i.e., "crunch" information, or rather, its representations (more details below);

- a mechanism for returning *output* that allows the device to react back to the environment, presumably with some effect (achieving some particular goal(s), learning, or survival, for example.)

Note the insistence on the generic word "mechanism" consistent with the desire for a scientific explanation (as opposed to the concept of a black box that just "gets the job done.") Mechanism is used here, for the time being, in a realistic sense and it may, or may not, imply a mechanic or electronic device. Brains excel as information processors despite the fact that many people may have difficulty with a brain being called "a mechanism."

These requirements may appear on the surface to be the standard abstraction of a processor, memory, and I/O that the classical Turing machine and von Neumann architecture give us and thus be nothing new. In view of the previous discussion, however, it is now clear that, as originally conceived in biological systems, these requirements have been essentially ignored in conventional models of computations because there are no specific I/O processes in Turing machines, finite automata, or formal grammars (more later.) As a result, the absence of this component has resulted in "missed opportunities" for models of computation, as pointed out by (Burgin, 2008). Although in abstract these features are appropriate for a general model of information processing, we must admit that an appropriate definition of computer science requires a fairly substantial reassessment of their nature, particularly concerning their implementation in the real world. In fact, we are proposing that the definition of computer science should entail giving as complete an explanation as possible of such models of information processing "mechanisms" by providing a less abstract and more grounded characterization of them in physical reality, albeit in a fundamental departure from the classical models, now along three fundamental directions: memory, adaptation, and evolution. We discuss them in turn. The changes are generally motivated by the idea that the devil is in the details, i.e., that the better direction to go is what 'might be termed "contextualization," "embodiment," and/or "complexification" rather than "abstraction," as discussed next.

1.3 Memory and States

The standard way to handle memory in computer science is to conecptualize it asn an abstraction of a physical representation of the information being remembered. The concept is known as a *state*. A state can assume many physical embodiments, and a more detailed discussion will be started in chapter 2 and revisited again in chapter 6. First, we naturally have to examine the concept of representation.

There is a phenomenon that implicitly pervades the computing field, but a satisfactory explanation of which has not been made central to the field and thus remains as yet unfledged in computer science. Despite substantial progress to elucidate how it actually works in other areas such as cognitive science, the phenomenon of memory surely holds the key to understanding a number of central issues in computing. In particular, questions such as "What does it mean 'to remember'?" and "How it is possible for humans to remember?" no longer admit a simplistic state-based explanation, especially in view of the fact that current storage-retrieval mechanisms used on computing machines perform so poorly when compared to human memory. Yet, memory seems to be an inherent property of a "brain," human or not. Life in general requires some sort of learning and adaptation, and it appears utterly impossible to argue that these capabilities are possible without some form of memory capability. Therefore any model of a computer must in some way or another involve some sort of memory system. In fact, a key feature of much of classical computer science (Turing machines, data bases) is that they make more sophisticated and appropriate choices for a solution as to the problem of memory. However, memories are not supposed to include means that automatically integrate them into the world, but require external intervention in choosing, rather imposing, the appropriate state set for a Turing machine, or schema for a database. The first departure in the appropriate definition of a computational model is therefore that it must include some "world model" in the form of a **functional** memory tightly integrated with the input/ouput coming from/to the real world. Turing machines come close to meeting this condition in abstract, but one must be very lax in accepting the abstraction of a linear tape or finite control as a valid real-world context. Neural nets come much closer in principle, although standard applications have been made in simulation in silicon (which were originally inspired and are still bound by the symbolic linguistic paradigm behind the computer metaphor.) Evolutionary systems likewise use relatively poor representations of the real world (symbolic data structures for genes and genomes), although they appear more tightly integrated to the world by bearing a closer resemblance to biological processes.

In general, we can distinguish three qualitatively different kinds of memory, corresponding to three stages of evolution, characterized by the key assumptions they make: symbolic, subsymbolic, and embedded. In symbolic computing, it is assumed all kinds of information can be expressed into finite strings of symbols from a basic finite alphabet of building blocks. Cognitively, a symbol is in general an empty box whose meaning is to be contextually interpreted by an observer in a specific interpretation of the symbol. The early bulk of the work in computer sciencereflected in current commercial products (conventional digital computers and software)is based on this type of representation, which has been the prevalent common assumption about data. Symbols are meant to be read sequentially, and so they are associated with sequential information processing, although symbolic representations are also used in processing by parallel machines. Sequential processing reflects the hard constraints of time and its passing, as it appears in ordinary experiences. By contrast, subsymbolic representations in the second kind are more distributed both in space and, hence, time. For example, the basic representation of heredity, the genome in a biological organisms, is a complex chain of basic DNA nucleotides a, c, g, and t; likewise, the physical manifestation of complex living organisms (protein ensembles) are known to be the genetic code of aminoacids, fairly complicated chemical compounds hardly reducible to simple character strings over an alphabet. More complex biological representations of information, such as organs and even brains, have a more complex structure. This type of extra-linguistic representation was achieved through experimentation in early physiology and psychology, and more recently in cognitive science and neuroscience. The representation of information here is by patterns of activation in the form of distributed representations across a large number of cells, each in itself only possessing a nearly negligible amount of memory and processing power. Eyes and brainsanimal and humanare primary processors of this kind. Examples of classical abstract models that process this kind of information are cellular automata, neural networks, and even analog and quantum computers. In the past, these processors were called massively parallel because they involved many more processors than sequential computation, although it is becoming clear now that they pale in comparison with natural systems such as eyes, brains, and even DNA molecules. The ontological status of symbolic and subsymbolic types of information is really an open question. Symbolic information is usually thought of as an abstract mathematical string or integer, while a neural activation level is conceptualized as requiring more complex quantities, perhaps requiring values in a continuum.

A third kind of information is expressed in and transformed by entire populations of separate individuals somewhere embedded in the real world. A genome (seen as a collection of chromosomes) is an example, but a zygote (fertilized biological ovum) is a better example. Its "information content" (one might dare say, its memory) originates and accumulates in context- and environmentally-dependent ways, often spread in a large number of cells across the population of cells and exhibiting a *prima facie* summary of the temporal sequence of events that formed it and put it there. It is transformed by processes of interaction of the individuals among themselves and with their environment. The individuals themselves undergo changes through evolutionary processes that adapt and shape them, but which appear unpredictable and entirely dependent on environments, populations, and their histories. The representation, usually called a genome, an organism, or even a society or culture, encodes only a number of key features that serve as a seed to the organism or device, but this representation needs to grow under various environmental conditions through a morphogenetic process into a fully operational information processing entity and, moreover, eventually decay into something else. The mature entity has usually generated a new genome that inherits most of its prior features, although some new ones may be incorporated in the process by genetic operations such as mutation and crossover with other individuals, or even, as recent evidence indicates, as a result of interaction with their environment in the same individual. These operations bear resemblance to biological processes, which were, in fact, the original inspiration of the concepts. Today, they go well beyond the ordinary biological interpretation.

The overall point, however, is that a substantial departure of the classical model means that these types of representation may not be exhaustive. Information may be represented and transformed in nature and the universe in a variety of other ways, some of which we are merely beginning to suspect. They involve such greater numbers of individuals than molecular densities localized in space that counting their number or quantifying their "magnitude" may be meaningless. For example, quantum computing is based on highly distributed representations known in physics as fields. Electromagnetic fields, for example light and electricity, are usually spread across space and change with time. A field at a point of space-time can be quantified by real numbers in some physical or mathematical model. But nothing prevents us from allowing a fragment of physical reality -such as a population of DNA molecules or subatomic particles in an entangled state in some isolated container- from being the model itself, even if we may not be in possession of a analytical physical, chemical, or biological model of that fragment of reality, as simple and desirable as it may be. The point is abstract state-based memories cannot be offered as a universal model of computation.

Let's discuss this type of representation in more detail bfore proceeding further in the next section.

1.4 Information Representation

The way information is stored and represented in a computer determines to a large extent its nature and capabilities. Books, CDs, human brains, sculptures, cities, and cultures are but examples of the many and diverse ways in which information is represented in nature. The representations and the transformations of information that they imply give rise to three types of computer models: conventional computers (modeled by Turing machines), connectionist networks (modeled by artificial neural networks), and evolware (genetic algorithms and genetic programming). Each expresses in different ways the hard constraints of time and space that are inherent to any information processing device, as we shall see. Therefore, every model of an information processing device is characterized by the *assumptions* it makes about information and how to process it. We can distinguish three qualitatively different kinds, corresponding to three stages of evolution, characterized by the key assumptions they make, as described in the following subsections.

1.4.1 Symbolic Representation

In symbolic computing, it is assumed that all kinds of information can be expressed into finite *strings* of symbols from a basic alphabet of building blocks. Here an *alphabet* is defined as an arbitrary finite set that contains a special symbol that is not visibly written (such as the 'blank' symbol). Cognitively, a symbol is in general an empty box whose meaning is to be contextually interpreted in a specific application of the symbol. The early bulk of the work in computer science, reflected in current commercial products (conventional digital computers and software), are based on this type of representation. This is the prevalent common assumption in many circles. For example, the basic representation of heredity, the genomes in biological organisms, are thus described as chains of symbols representing the basic DNA nucleotides A, C, G, T. Symbols are meant to be read sequentially, and so they are associated with sequential information processing, although symbolic representations are also used in processing by parallel machines. Sequential processing reflects hard constraints of *time* and its passing, as it appears in ordinary life. Sequential models will be explored in that order in chapters 2 thorugh 6.

We illustrate the standard representation with the binary alphabet $\mathbf{B}=\{0,1\}$. Its elements can be regarded as the nodes of a rooted infinite binary tree. The root is the *empty* word consisting of no symbols, hereforth denoted λ. A given word at a node has two children, the left one obtained by *concatenating* a 0 on the right and a right child obtained by catenating a 1, also on the right. Full recursion of these operations from the root yield the complete binary tree representing all words over $\mathbf{B}$. *Each word only has a finite number of symbols, although there is an infinite number of words.* A similar tree can be built for words over the English alphabet (52 upper and lower case letters plus additional special symbols). These symbols are written for humans just about everywhere,

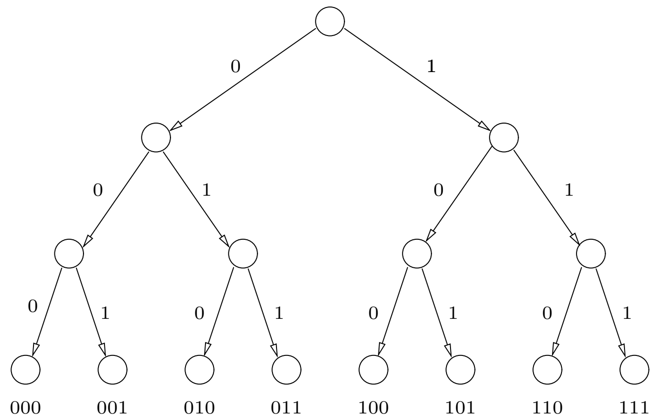

Fig. 1.1. Words over the binary alphabet.

except that they are 'folded' into a book, a CD, or an electronic file, for practical reasons. The abstract symbols take on a specific meaning when they become long strings of this sort.

Another type of representation in biology can be captured at the symbolic level, although perhaps barely. Biological representations of genetic information can be regarded as symbolic expressions over a genetic alphabet possessing a more complex structure. The alphabet now contgains four symbols $\mathbf{D}= \{a, c, g, t\}$ (the bases) representing complex molecules that assemble in single and double chains to encode genetic information. The single strands are formed through co-valent bonding (of the kind that binds together hydrogen and oxygen to form water) that concatenates basis units into chains that can be very roughly described by strings over this genetic alphabet. **D** is much richer in the sense that these single strands need to be further tied together by a weaker hydrogen-based bonding to form the familiar double stranded DNA helix through an separate process called *Watson-Crick complementation* (WC), see Fig. 1.2. WC binds together a's and t's, as does c's and g's in alignments between the two single strands determined globally by their so-called Gibbs energy released in the process. Further actions by enzymes present in the living cell (topoisomerases) twist the double strands billions of base pairs long into a very compact chromosome that contains all the information necessary to reproduce the organism they belong to. Double strands are thus distinguished from ordinaray strings in that, in addition to a polarity (sense of direction, from the so-called $5'$ to $3'$ end), they possess a higher order structure (induced by WC complementation and twisting) that is critical to their structure and function.

Fig. 1.2. DNA strands over a 4-letter alphabet.

1.4.2 Subsymbolic Representation

More recently we have come to realize, through the development of physiological technology and experimentation, cognitive science, and connectionism, that superior information processing capabilities probably rely on extralinguistic representations and factors even more hardly describable in words. Human faces, natural scences, segments of real life, or even are examples of such. A more natural representation of information is by *patterns of activation* in the form of distributed representations across a large number of cells, each in itself only possessing a relatively small amount of memory (usually one of a pallette of colors, each referred to as a *pixel*). These representations are associated with information processing by parallel machines, and although still representable by symbols, the semantic meaning associated with the representation of a fixed concept requires a lot more pixels than symbols in the sequential case. Eyes and brains, animal and human, are primary processors of this kind. Examples of models that process this kind of information are cellular automata, neural networks and even analog computers. These models will be covered in Chapter 6, and in a different way, also in chapters 7-9. In the past, these processors were called massively parallel because they involved many more processors than sequential computation, although it is becoming clear now that they pale in comparison with natural systems such as eye, brains, and molecules of DNA.

An important feature of this type of representation is that, usually, it takes values over a continuum, and therefore a single instance may require an infinite sequence of symbols when expressed as expansions (decimal or binary). There is a sharp contrast with the symbolic representation, where a symbol variable can have only relatively few values (perhaps a few thousand in the case of chinese characters) that take only a finite amount of time and space to express.

1.4.3 Genetic Representations

A third kind of information is expressed in and transformed by entire populations of individuals. DNA is but an example. The information originates and accumulates in context- and environmentally-dependent ways, often spread in spaces across all individuals of the population. It is transformed by processes of interaction of the individuals with others and their enviroment. The individuals themselves are changed through evolutionary processes that adapt and shape them but appear unpredictable and entirely dependent on enviroments, populations and their histories. The representation, usually called a *genome*, encodes only a number of key features that serve as a seed to the organism or device, but this representation needs to *grow* under various environmental conditions to a fully operational information processing entity. The mature entity then generates a new genome that inherits most of its prior features, although some new ones may be incorporated in the process by genetic operations such as mutation and crossover with other individuals. These operations bear ressemblance to biological processes, and were, in fact, the original inspiration of the concepts, but today they go well beyond the ordinary biological interpretation. These topics are covered in chapters 7 and 8. Chapter 7 studies for the first time a model that is tightly anchored in the Darwinian paradigm of natural selection, or survival of the fittest. These model realize the massive parallelism in nature envisioned in sybsymbolic systems, at greater times scales. For example, a test tube can typically contain about 10^{18} individual processors (nucleotides that also constitute the basic building block of human genomes) in a volume of about a cubic centimeter (under sixteenth of a cubic inch).

Information is represented and transformed in nature in a variety of other ways, some of which we are merely begining to suspect. They involve such greater numbers of individuals than molecular densities that counting their number becomes almost meaningless. For example, quantum computing is based on highly distributed representations known in physics as *fields*. Electromagnetic fields, for example light and electricity, are usually spread across space and change with time. To quantify a field at a point of space-time requires real numbers. A single real number generally requires an infinite number of symbols. Some inroads to this type of information processing are discussed in Chapter 9. They include information processing in physical systems, including the possibility, for example, that the most fundamental particle in the universe may be an *infon*. Energy, and even physical objects, would then be derived quantities, different manifestations of information. Organizing data and information about information into coherent knowledge is part of our undertaking.

1.5 Recognition, Generation, and Compaction

In the previous sections we have presented the corner-stone theme, information. This basic concept runs through the heart of computer science, generating a

great deal of problems, and also serving as a unifying thread while spawning great a number of problems. To handle them, a very useful approach in science consists of identifying **fundamental questions** that
a) come up again and again, often in a variety of desguises, and
b) pose challenges whose solutions afford great leaps in the development of the field.
Physics, for example, developed enormously in the process of uvelining the basic conundrum of planetary motion in astronomy. Computer science is no exception. Taking this approach, four problems stand out as the most fundamental, among other things because most problems in information processing can be ultimately reduced into one of these categories, described next.

- *The Recognition problem*

  One of our most amazing abilites is to recognize and categorize objects in nature, particularly humans faces. Whenever we see someone, it usually takes but a couple of seconds to make a decision whether the face belongs to friend or foe. This is an example of a recognition problem. Each of us has met a different number of faces in our lifetime. The problem is to decide whether or not a random new face presented to use belongs to that ensemble. Of course, if the decision is negative, me may change the ensemble and be able to recognize it as positive at the next presentation, but this is still the same problem with a different new ensemble.

  In terms of information processing, the recognition problem is always about a given ensemble of objects E. The recongition problem of ensemble E is **to devise a procedure/mechanism to recognize whether an** ***arbitrary*** **object belongs to the ensemble** E **or not**. A typical example occurs in compiler design, where the given objects are arbitrary input "programs" (good and bad), and the compiler needs to decide if the program satisfies the constraints imposed by the language specification. A further example is recognizing grammatical sentences in a natural language such as English. Human brains are clearly solutions to these problems, but we cannot build them at will because we don't understand yet how they work. We want solution we understand and can build. Solutions of this problem are automata, neural nets both natural and artificial, and morphological detectors such as those present in the immune system. We will discuss them in detail later.

- *The Generation problem*

  Likewise, given an ensemble of objects E, he recongition problem of E is **to devise a procedure/mechanism to produce** *all and only* **the special objects in the set** E. In the case of human faces, we now rather want a printout of all faces have ever seen at a particular moment. In computer science, a typical example occurs when specifying a programming language, where it is important to define very precisely the strings that define legal/valid programs. Known solutions to this problem are Chomsky grammars, Markov Algorithms, Post systems, and splicing systems.

Another solution of a biological type is a morphogenetic process that allows the genome of an individual to be expressed into a new organism through a developmental process. Unfortunately, this type of solution has not captured yet the same degree of attention that genomes and brains have in traditional computer science, although it is beginning to be recognized as an important process in fields such as evolvable hardware and the 11'omics" in bioinforamtics (such as genomics and transcriptomics).

- *The Compaction problem*
  The compaction problem is another problem of paramount importance in computer science. Given an ensemble of objects or their descriptions, it calls for a compact way to represent the same amount of information with minimum requirements on the amount of storage space as well as the amount of time required to retrieve or reconstruct them. As we will see time and time again, this problem has enormousimportance for storage and communication. Typical examples are video compression for broadcasting, the human problem of storing lots of faces in a relatively small human brain, and the problem of storing the shape and composition of a human creature inside nanoscopic DNA strands in the interior of living cells. A typical solution is given by several compaction algorithms such as Hoffman coding and Zvi-Lempel. An example of a different kind is human memory. Humans can store amazingly large amounts of information in extremely compact and fault-tolerant ways when compared to ordinary computer memories. The dutch philosopher Blaise Pascal put it well in saying (paraphrased): "The universe holds me physically; with my mind, I encompass it all." Some people hold the opinion that human memories require a great deal of reconstruction, as opposed to the kind of relatively simple retrieval that typically occurs in electronic memories. A genome is yet another example, where the storage is so implicit that much of the encoding and decoding takes place through a complex process of reproduction and morphogenesis in an appropriate environment.

- *The Programming problem*
  Given a specific type of transformation on a body of data (say, the current atmospheric conditions and the record of the weather in the last century), and given a computing device of a particular sort (say, a personal computer with a specific instruction set), assemble the right combination of instructions from the basic set to achieve the desired transformation (e.g., a weather forecast for the next day.) This is a problem that has proved particularly difficult for electronic computers. Solving it requires extensive design and analysis of algorithms, writing software implementations in a particular language, and, testing, validating, installing and maintaining it as the computing devices available continously evolve. The desired transformations of information may be enormously varied and complex, such as weather forecasting, maintaining inventory for a world-wide corporation, or examining the climate on planet earth in 100 years to determine

> whether there is indeed global warming. Here, we will only be concerned with the problem of determining whether the software can be written, how inherently difficult it would be, and how to prove its correctness. Systemic issues such as installation, evaluation and maintainance will require other techniques from fields such as as software development and software engineering.

As will become clear through the following chapters, the types of solutions given to these problems characterize the various stages of development in computer science.

1.6 Environment and Input/Output

In addition to memory and information representation, the second essential ingredient in a computing device is some form of input/output. After all, a computing device needs to be given, or produce by itself, in some way or other, the description of the task(s) to be performed. The obvious way to communicate it is by using *strings* over a fixed alphabet, the way humans communicate in writing with the help of a standard set of symbols (for example, the roman characters plus the numerical and special symbols in the English alphabet). Many other ways naturally occur in the animal kingdom, however, even in verbal communications involving facial expressions and body language. These nonlinguistic expressions convey a great deal of information content, if not all of it (in being ironic, for example).

Even a complete description of a genome will not, in general, give us an idea of how to build the corresponding organism (e.g., a human), let alone understand its information processing abilities. In molecular computing, for example, "DNA is a dead molecule, amongst the most nonreactive, chemically inert molecules in the living world. [$\cdots$]Only a whole cell may contain all the necessary machinery for self-reproduction. [$\cdots$] A living organism at any moment in its life is the unique consequence of a developmental history that results from the interaction of and determination by internal and external forces [$\cdots$]" (Lewontin [BN, p. 200]). Genetic material needs to be expressed, and the expression depends substantially on the environment in which it takes place. A crucial difference in evolutionary approaches to computation lies in accepting the fact that systems should not be entirely predetermined as independent entities, but rather should be left open to change until run time so that context and *environmental* conditions can be accounted for. In particular, emphasis in programming should be placed in identifying a set of interaction constraints, as opposed to abstracting or simulating details of reality in an artificial environment. The bulk of the computation lies in the *process* of exchange with the environment, i.e., input/output, not in following a fixed set of rules insensitive to the situation where the computing device finds itself. Thus, programs interact with environment in an evolutionay processes that adds fundamental new components through learning, self-modification, automatic programmming, and evolution. This is one key feature that the new field of *interactive computation* captures. This aspect of computation shades a great deal of doubt into the symbolic hypothesis, but its study also enlightens the understanding of computation, as will be seen in the following pages.

1.7 Complexification Processes

For centuries, traditional sciences (including computer science) have strived to identify key components in their objects of study in a reductionistic program that is supposed to reduce complex to phenomena to a few parameters governed by simple rules, usually expressed in a single equation or formula, such as E=Mc2. They are attempts to find *simple models of complex phenomena*, such as chess playing or deductive reasoning. In constrast, much of the contribution of natural selection and evolution, and a key component in evolutionary strategies and a new AI (Artificial Intelligence), is to develop *complex models of apparently simple phenomena* such as perceiving or walking, in an attempt to understand how complexity and information builds up in us and the processes around us. Thus far, most of the complexification process remains essentially ignored in information processing, although it is obviously one of the most important aspects in biology and life processes. *Nature vs nurture* is an important duality in the understanding of life. It is also becoming increasingly important in the understanding of information processing machines as well, as we will discover in the following pages.

1.8 Computation Tomorrow

We thus come to the current state of affairs in the computing field. As mentioned above, the central goal of computer science is to provide inspiration to build and understand, including reasonable predictions about the type of, information processing machines now and in the future. Most efforts have been focused on the transformation of data or information, as opposed to the nature of information itself. The impact of technology, as usual, may end up having a decisive effect on the fundamental problems that computer science should be concerned with. One can be almost certain, however, that the following problems will remain fundamental because they appear deep, difficult, and central to computer science.

- *What is information?*
  It is crucially important to distinguish information from its embodiment, usually called *data*, as well as information processing from *data communication*. Thus Shannon's Information Theory is not appropriate to answer the fundamental questions about information. A case can be made that information can be organized in specific ways for specific purposes, in which

form it is sometimes called *knowledge*. But an empirical taxonomy of that sort does not answer the question, in the same way that behaviorism does not present an explanation of the phenomenon of *cognition*, but rather a body of facts that call for an explanation, theoretical or otherwise.

- *How can information be efficiently represented?*
  It becomes clear upon observing natural phenomena that nature encodes and represents information in ways far more efficient than we do, as exemplified recently by Adleman's natural solution of a family of problems (The Traveling Salesrep Problem) that appears extremely difficult for conventional computers (more in Chapter 8).

- *How is information created, transduced and recoded by natural processes?*
  It is increasingly clear that living entities create information critical for their survival and success in life. Whether or not the underlying natural processes admit the kind of analysis and description by laws of information processing of comparable significance with known physical and biological principles, or whether it will require laws of an entirely different nature, remains one of the great mysteries of computer science and even biology.

Finally, there is the question of how these questions relate to similar questions in increasingly related fields, such as cognitive psychology, biology, and artificial intelligence. A lot of the approach taken in this book to computer science is motivated by how closely the problems listed above resemble analogous problems faced by psychologists and biologists when facing the phenomenon of information processing by biological organisms. See, for example, Varela-Thompson-Rosch [VTR]. Some people do not even see a sharp distinction between the two sets of questions. Some current research lends credence to the belief that they are, in fact, two aspects of the same problems and likely will have similar solutions. We will keep these connections in mind throughout, while remaining focused on our computational theme.

1.9 Summary and Conclusion

The fundamental object of computer science is the study of information, its organization and processing by means of devices that can be actually built or found in nature. We have reviewed the most important issues that this enterprise entails, as well as mentioned some of the solutions to be studied in detail in the following chapters. The fundamental questions to be addressed are:
What is a computer?
How can information be represented and processed efficiently?
What is information?
The key tools in the answers to these questions to date are models known as finite-state machines, Turing machines, cellular automata, neural networks, genetic algorithms, and molecular computers. These tools and their relationships

to analogous questions in biological and psychological phenomena constitute the theme of the following pages.

1.10 Problems

These sections contain a list of exercises and problems that allow the reader to confirm understanding of basic concepts in each chapter, as well as to see applications, experiments to perform, and perhaps open questions worthy of further inquiry (either in the literature*, or on his own**). In this first chapter, they will take the form of thought-provoking questions to ponder, rather than specific problems to work out an answer for.

INFORMATION REPRESENTATION

1. Discuss the distinction, or lack thereof, between ‘information content’ and ‘information representation’.

2. Does the environment determine how information gets represented in a digital computer system? In a biological organism?

3. Can the environment encode information? If so, can environmental changes be regarded as information processing? Do digital computers perform operations of this sort? How about biological organisms?

4. Discuss the feasibility of having an alphabet with an infinite number of symbols for the purpose of information representation. [Can a *glimpse* be an atomic unit of representation and visual processing, for example?]

5. Discuss the plausibility that symbols are appropriate to handle mental representations and thinking in information processing by humans.

6. Name and discuss three nonlinguistic forms of information representation.

7. Design a way to represent numbers in binary square arrays so that addition and multiplications can be performed more easily on them than with linear representations (possibly by operating on their pixels in parallel).

INFORMATION

8. What exactly is information? *,**

9. Does information have observer independent existence? **

10. Is there a minimal unit of information? Do *infons* exist? **

INPUT/OUTPUT

11. What is the difference between *sensation* and *perception*? *

12. What should be the proper role of *perception* in the treatment of information processing?

13. Do conventional computers have any type of sensation? Perception? Should they be endowed with such properties?
    What are the risks involved?

    COMPLEXIFICATION *,**

14. Do computers create data?
    Do they process data?
    Do they create information?
    Do they process information?
    Do they create knowledge?
    Do they process knowledge?
    Do living organisms create and/or process information?
    Do animals create and/or process knowledge?
    To what extent does a genome determine completely a living organism?
    What is the role of environmental conditions on information processing by living organisms?
    How could computers be enhanced to create and process information?
    What are the risks involved?
    What is the ultimate source of information?

Notes

These sections contain brief important historical observations about the subject of the chapter, how it relates to previous chapters and subjects, a summary of success stories, e.g., more important applications, as well as how they seem to relate to the fundamental problems discussed in this chapter. Includes particularly worthy items for further reading from the bibliography.

References

[] This section contains a selected list of references for the chapter (perhaps annotated).

[Ba] D. Ballard [1997]. *An Introduction to Natural Computation.* The MIT Press, Cambridge, MA

[BN] K. Dewdney [1989]. *The New Turing Omnibus.* W.H. Freeman and Company, New York, NY.

[D] K. Devlin [1991]. *Logic and Information.* Cambridge University Press, Cambridge, UK

[De1] A.K. Dewdney [1989]. *The New Turing Omnibus.* W.H. Freeman and Company, New York, NY.

[De2] A.K. Dewdney [1996]. *Introductory Computer Science (Bits of Theory, Bytes of Practice).* Computer Science Press.

[HU] J. Hopcroft, J. Ullman [1979]. *Introduction to automata theory, languages and computation* Addison-Wesley, Reading, MA

[Mi] M. Minski [1967]. *Computation (finite and infinite machines).* Prentice-Hall, Englewood Cliffs, NJ

[VTR] F.J. Varela, E. Thompson, E. Rosch. [1991]. *The embodied Mind: Cognitive Science and Human Experience.* The MIT press, Cambridge, MA.

[URL] COMPUTER SIMULATIONS
The following are email addresses providing packages to simulate many of the models studied in the following chapters, or information where to obtain such packages:

- `http://www.csd.uwo.ca/research/grail/links.html`
`grail` is a suite of algorithms to handle finite-state machines. Other such packages are `Amore`, `FIRE Lite`, `automate`. They are usually written in a C-like language and are fairly portable.

- `http://www.cs.duke.edu/~rodger/tools/tools.html`
includes simulation of standard symbolic models, from finite-state machines to Turing machines, over the web.

- `http://www.it.uom.gr/pdp`
is a digital laboratory equipped with resources on parallel and distributed computing. It includes facilities for artificial life, neural nets, genetic algorithms and so forth.

- `http://iicm.tu-graz.ac.at`
Similar to the previous one, although from a more algebraic point of view.