# 1 Introduction to Informatics

## 1.1 Basics of Informatics

*Informatics* is a very young scientific discipline and academic field. The interpretation of the term (in the sense used in modern European scientific literature) has not yet been established and generally accepted [1]. The homeland of most modern computers and computer technologies is located in the United States of America which is why American terminology in informatics is intertwined with that of Europe. The American term computer science is considered a synonym of the term informatics, but these two terms have a different history, somewhat different meanings, and are the roots of concept trees filled with different terminology. While a specialist in the field of computer science is called a computer engineer, a practitioner of informatics may be called an informatician.

The history of the term computer science begins in the year 1959, when Louis Fein advocated the creation of the first Graduate School of Computer Science which would be similar to Harvard Business School. In justifying the name of the school, he referred to management science, which like computer science has an applied and interdisciplinary nature, and has characteristics of an academic discipline. Despite its name (computer science), most of the scientific areas related to the computer do not include the study of computers themselves. As a result, several alternative names have been proposed in the English speaking world, e.g., some faculties of major universities prefer the term *computing science* to emphasize the difference between the terms. Peter Naur suggested the Scandinavian term *datalogy*, to reflect the fact that the scientific discipline operates and handles data, although not necessarily with the use of computers. Karl Steinbuch introduced in 1957 the German term *informatik*, Philippe Dreyfus introduced in 1962 the French term *informatique*. The English term *informatics* was coined as a combination of two words: *information* and *automation*; originally, it described the science of automatic processing of information. The central notion of informatics was the *transformation of information*, which takes place through computation and communication by organisms and artifacts. Transformations of information enable its use for decision-making.

| | |
|---|---|
| **Q&A**<br>What is informatics? | **Technical definition:** Informatics involves the practice of information systems engineering, and of information processing. It studies the structure, behavior, and interactions of natural and artificial systems that collect, generate, store, process, transmit and present information. Informatics combines aspects of software engineering, human-computer interaction, and the study of organizations and information technology; one can say it studies computers and people. In Europe, the same term informatics is often used for computer science (which studies computers and computer technologies).<br>**Business definition:** Informatics is a discipline that combines into one-field information technologies, computer science and business administration. |

A number of experts in the field of computer science have argued that in computer science there are three distinct paradigms. According to Peter Wegner, these three paradigms are science, technology and mathematics. According to Peter J. Denning, these are theory, modeling and design. Amnon H. Eden described these as rationalist, technocratic, and scientific paradigms. In his opinion, in frames of rationalist paradigm the computer science is a branch of mathematics; mathematics dominates in theoretical computer science and mainly uses the logical conclusion. The technocratic paradigm is the most important in software engineering. In the frame of a scientific paradigm, computer science is a branch of empirical science, but differs in that it carries out experiments on artificial objects – software and hardware.

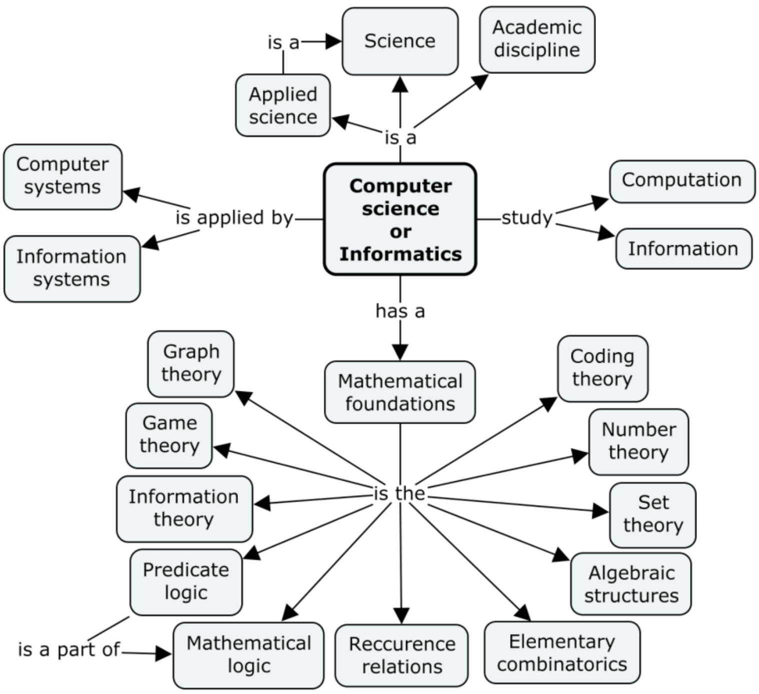


**Figure 1:** Foundations of computer science and informatics.

An overview of computer science can be borrowed from Charles Chen [2]. He referred to mathematical, theoretical, and practical (application) branches as key components. The mathematical branch is devoted to systems modeling and creating applications for solving math problems. Some related principles of study are classical and applied mathematics, linear algebra and number theory. The theoretical branch covers algorithms, languages, compilers and data structures. This branch is based on numerical and logical analyses, discrete mathematics, statistics, programming theory, software and hardware architecture. The practical branch contains operating systems, different apps, frameworks, and software engineering. Real world applications of computer science include cryptography, software development, project management, IT system support, artificial intelligence, and computer graphic.

The dominated meaning of the term *informatics* has changed in different periods of development of the science. At first, informatics was treated as a part of library science – the theory of scientific information activity. This means practical work on the collection, analytical and synthetic processing, storage, retrieval and provision of scientific information laid down by scientists and engineers in their documents. Informatics was a discipline that studied the "structure and general properties of scientific information, as well as the laws of its creation, transformation, transmission and use in various spheres of human activity". Later, it turned more in the direction of the science of computers and their applications and informatics came to mean a computing and programming discipline engaged in the development of the theory of programming and of application of computers [3]. The last meaning of the term as fundamental science of information processes in nature, society and technical systems was coined in the early 1990s. According to this interpretation informatics examines the properties, laws, methods and means of formation, transformation and dissemination of information in nature and society, including by means of technical systems. The latter change caused that besides informatics specializations like theoretical computer science or technical informatics there are also social informatics, biological informatics, and physical informatics. Nowadays we can observe the coexistence of all three definitions of the word "informatics". This fact complicates the study of this scientific direction.

Informatics has evolved in eight major areas:

1. Theoretical informatics
2. Cybernetics
3. Programming
4. Artificial Intelligence
5. Information systems
6. Computing equipment
7. Social informatics
8. Bioinformatics

### 1.1.1 Historical Background

The history of informatics [4], in the modern sense, begins in the middle of the XX century, due to the advent of computers and the beginning of the computer revolution. Informatics or computer science is only now entering the age of early maturity. Some historians date the beginning of the informatics era from the date of the creation of the first electronic computer. Others prefer to date it from the first electric (and even mechanical) calculators. As it always happens in such situations, there are many opinions and justifications for each point of view. If we take 1950s as the beginning of the history of modern informatics, all previous events (important in the context of information processing science) are considered prehistoric. There were some very important inventions and discoveries during prehistory, which allow us to trace the historical logic of the creation of modern information technologies.

To understand the roots of informatics, one should look at the history of computer technology [5], which includes a multitude of diverse devices and architectures. The abacus from ancient Babylon (300 BC) and China (500 BC) are the oldest known historical examples. The Jacquard loom invented by Joseph Marie Jacquard (1805) and the analytical engine invented by Charles Babbage (1834) are the first examples of zero generation (prehistoric) computers – mechanical machines designed to automate complex calculations. De facto, Babbage's engine was also the first multi-purpose programmable computing device. After him, George and Edvard Scheutz (1853) constructed a smaller machine that could process 15-digit numbers and calculate fourth-order differences. Ada Lovelace (1815-52) collaborated with Charles Babbage. She is said to be the first programmer. She saw the potential of the computer over a century before it was created. It took more than one century, until the end of the 1960s, when mechanical devices (e.g. the Marchant calculator) found widespread application in engineering and science.

First generation electronic computers (1937-1953) used electronic components based on vacuum tubes. This was the period when both digital and analog electric computers were developed in parallel. The first electronic digital computer was ABC, invented by John Vincent Atanasoff at Iowa State University. The second early electronic machine was Colossus, designed by Alan Turing (1943) for the British military. The first general purpose programmable electronic computer was ENIAC (1943-1945), built by J. Presper Eckert and John V. Mauchly at the University of Pennsylvania. In fact, as late as the 1960s, analog computers still were used to solve systems of finite difference equations. Nevertheless, digital computing devices won the competition, because they proved to be more useful when dealing with large-scale computations (more computing power, more scalable and economical). Software technology during this period practically did not exist; programs were written out in machine code. Only in the 1950s a symbolic notation known as assembly language, began to be used.

Second generation computers (1954-1962) were based on semiconductor elements (discrete diodes and transistors) with a very short switching time and randomly accessed magnetic memory elements. There was a new opportunity to perform calculations in the format of real numbers (floating point). High-level programming languages such as FORTRAN, ALGOL and COBOL were created during this time. These changes enabled the production of the first computers designed not only for science and the military, but also for commerce. The first two supercomputers (LARC and IBM

7030) were designed during the same period. These were machines that had much more calculating power than others and could perform parallel processing.

The key changes in the third generation of computers (1963-1972) concerned the use of integrated electronic circuits instead of discrete elements and the use of semiconductor memory instead of magnetic cores. Changes in computer architecture were associated with the spread of operating systems and parallel processing techniques. These developments led to a revolutionary increase in the speed of execution of calculations. The key figure for the development of computer technology in this period was Seymour Roger Cray who established the supercomputer industry through his new architecture approaches. His computers attained for the first time a computation rate of 10 million floating-point operations per second (10 Mflops). At the same time (1963) Cambridge and the University of London developed in cooperation Combined Programming Language (CPL), which became the prototype for a number of other major programming languages. Early implementations of the excellent operating system – UNIX were based on the language B, which is derived from the CPL.

The next (fourth) generation of computer systems (1972-1984) went through the use of large and very large scale integration of electronic components on chips. This has enabled an entire processor and even an entire simple computer to fit onto a single chip. That very important change allowed for a reduction in the time required to execute basic calculations. Most systems acquired a semiconductor main memory, which also increased the speed of information processing. There were two very important events at the beginning of the fourth generation: the development at Bell Labs of the C programming language and the UNIX operating system (written on C). In a short time, UNIX was ported to all significant computers; thousands of users were exempted from the obligation to learn a new operating system each time they changed computer hardware. The new trend in software methodology during the fourth generation was the introduction of a declarative style of programming (Prolog) next to an imperative style (C, FORTRAN).

The fifth generation of computer technology (1984-1990) was characterized mainly by the widespread adoption of parallel processing. This means that a group of processors can be working on different parts of the same program. At that time, fast semiconductor memories and vector processors became standard on all computers, this was made possible by rapid growth of the scale of integration; in 1990, it was possible to build integrated circuits with a million components in one chip. Thus, all the processors participating in parallel calculations may be located on the same PC board. Another new trend was the common use of computer networks; connected to the network, single-user workstations created the conditions for individual work at the computer in contrast to the previous style of work in a group.

The latest, i.e. the sixth generation of computers (1990- ) is distinguished by the exponential growth of wide-area networking. Nowadays, there is a new category of computers (netbooks), which are useful only when using the Internet. Another category, mobile computing devices (smartphones, palmtops, tablet PCs) are strongly dependent on the network. The computer spread out from science, technology and industry to all areas of our lives.

### 1.1.2 Famous People in the History of Computing

Many people shaped Informatics into what it is now [6]. We all know those who have had an immense influence on the industry, like *Bill Gates* or *Steve Jobs*. However, let us take a look at some lesser known individuals.

*Grace Murray Hopper* is the author of term *debugging* (1947), which means detecting and removing errors from a computer program. He is one of the originators of the idea of machine-independent programming that led to the development of modern programming languages starting with COBOL.

*Ken Thompson* and *Dennis Ritchie*, two engineers at AT&T's Bell Labs research invented OS UNIX (1969). From the beginning, it was a multitasking, multiuser computer operating system; all Apple's personal computers still owe their success to this operating system.

*Seymour Cray*, who has already been mentioned, set the standard for modern supercomputing. Two important aspects to his design philosophy were removing heat, and ensuring the electrical length of every signal path on board. That is why all his computers were equipped with efficient built-in cooling systems, and all signals were precisely synchronized.

*Marvin Lee Minsky*, co-founder and supervisor of Artificial Intelligence Laboratory (AI Lab) at Massachusetts Institute of Technology (MIT) is the author of foundational works on the analysis of artificial neural networks. He has made many contributions to not only AI, but also to cognitive psychology, computational linguistics, robotics, and telepresence (the use of virtual reality technology for remote control and for apparent participation in distant events).

*Douglas Carl Engelbart* is generally well known for inventing the computer mouse (1960s). More specifically, he developed some of the key technologies used in computing today. First, he advanced the study of human-computer interaction that resulted in developing desktop user interface and other components of graphical user interfaces (GUI). His efforts led to the development of hypertext and networked computers too.

The originator of the term *artificial intelligence,* computer scientist *John McCarthy* developed the LISP (1958) – the second high-level programming language in the history of computing; to this day, programmers use Lisp dialects. In the 1960s, McCarthy intensively popularized sharing a computing resource among many users; which means the use of multi-tasking (concurrent processes) and multiprogramming approach to building software systems.

Software designer *Tim Paterson* is the original author (1980) of MS-DOS; de facto, Bill Gates only rebranded his operation system known at first as QDOS (Quick and Dirty Operating System). Paterson worked for Microsoft on MSX-DOS and on Visual Basic projects.

*Daniel Singer* and *Robert M. Frankston* are the creators of the first spreadsheet computer program *VisiCalc*. It was historically the first program that has transformed microcomputers from a plaything into the professional tools needed to perform specialized calculations.

*Robert Elliot Kahn* and *Vinton Gray Cerf* are the inventors of two basic communication languages or protocols of the Internet: TCP (Transmission Control Protocol) and IP (Internet Protocol). Both, TCP and IP are currently in widespread use on commercial and private networks.

*Niklaus Wirth* is known as the chief designer of several programming languages, including Pascal. He has written very influential books on the theory and practice of structured programming and software engineering.

*James Gosling* invented (1995) the Java programming language, which become popular during the last decade. Compiled Java applications typically run on Java virtual machines regardless of computer architecture. Independence from the computer architecture has been achieved in such a way.

The inventor of the World Wide Web (WWW, an internet-based hypermedia initiative for global information sharing), *Timothy John Berners-Lee* implemented in 1989 the first successful client-server communication via the Internet using Hypertext Transfer Protocol (HTTP). Nowadays he is the director of the W3C – the main international standards organization for the WWW.

*Linus Benedict Torvalds* created Linux – the kernel of the operating system GNU/Linux, which is currently the most common of free operating systems. Currently, only about two percent of the current system kernel has been written by Torvalds, but he remains the decision maker on amending the official Linux kernel tree.

It is not possible to mention here all who have contributed to the development of modern computer technology. There have been, and will continue to be, many others.

### 1.1.3 Areas of Computer Science

Computer science combines a scientific and practical approach. On the one hand, computer scientists specialize in the design of computational and information systems, on the other hand they deal with the theory of computation and information. To understand the areas of interest of computer scientists, we must analyze from where they or their teachers came to computer science: from electronics, from mathematics, from physics and so on.

In 2008 ACM Special Interest Group on Algorithms and Computation Theory published its vision of theoretical computer science themes that have the potential for a major impact in the future of computing and which require long-term, fundamental research. These are the issues of algorithms, data structures, combinatorics, randomness, complexity, coding, logic, cryptography, distributed computing, networks among others. Of course, some central research areas were not represented here at all, but vision can help us to understand the field of computer science theory.

Let us browse theoretical and applied areas, which are relatively stable and are represented in computer science curricula. Generally, theoretical sciences form a subset of wide-ranging computer science and mathematics. This is why all of them focus on the mathematical aspects of computing and on the construction of mathematical models of different aspects of computation. Someone who loves mathematics, can find among these disciplines his/her preferred area of interests. Applied sciences are directly related to the practical use of computers. The following lists contain names of the most popular theoretical and applied disciplines belonging to area of computer science:

| **List 1. Theoretical computer science** | **List 2. Applied computer science** |
|---|---|
| Algorithms and data structures | Artificial intelligence |
| Algorithmic number theory | Computer architecture and engineering |
| Computer algebra or Symbolic | Computer Performance Analysis |
| computation | Computer graphics and visualization |
| Formal methods | Computer security and cryptography |
| Information and coding theory | Computational science |
| Programming language theory | Computer networks |
| Program semantics | Concurrent, parallel and distributed systems |
| The theory of computation | Databases |
| Automata theory | Health informatics |
| Computability theory | Information science |
| Computational complexity theory | Software engineering |
| Formal language theory | |

### 1.1.4 Theoretical and Applied Informatics

Before the end of 1970s, cybernetics was the dominant term among sciences related to the processing of information; respectively, theoretical informatics was named *mathematical (theoretical) cybernetics*. Briefly, theoretical informatics is a mathematical discipline, which uses methods of mathematics to construct, and study models of processing, transmission and use of information. Theoretical informatics creates the basis on which whole edifice of informatics is build. By its very nature, information tends to discrete representation. Data messages typically can be described as a discrete set. This is why theoretical informatics is close to discrete mathematics, and many models of theoretical informatics are borrowed from discrete mathematics. However, as a rule, these models are filled with content related to the particulars of information as an object of study.

#### 1.1.4.1 Theoretical Informatics

Theoretical computer science itself is divided into a number of distinct disciplines. According to tasks clusterization, theoretical informatics can be divided into five areas:

1. Information disciplines based on mathematical logic. These develop methods to use the achievements of logic to analyze the processes of information processing using computers (e.g., *algorithms* or the *theory of parallel computing*). As well, they deal with methods by which it is possible to study the processes taking place in the computer during computation (e.g., *automata theory* or the *Petri net theory*).
2. *Theory of computation*. Previously, mathematicians do not care about the possibility of transferring their methods of solving problems in a form that allows the programming. The expansion of computers stimulated the development of special problem-solving techniques in mathematics, so disciplines lying at the boundary between discrete mathematics and theoretical computer science (e.g., *computational mathematics and geometry*) emerged.
3. *Theory of Information*. This means the study of information per se (i.e. in the form of an abstract object, devoid of specific content). Theory of information engages the general properties of information laws that govern its birth, development and destruction. This science is closely related to *coding theory*, whose mission is to study the forms in which the content of any particular information item (e.g. the message being sent) can be “cast”. In information theory, there is a section specifically dealing with theoretical issues, which concern transmission of information through various channels of communication.
4. *System Analysis*. Informatics has to deal with real and abstract objects. The information circulating in the real form is materialized in various physical processes, but from a scientific perspective, it acts as an abstraction. This shift causes the need to use on computers the special abstract (formal) model of the physical environment in which the information is perceived in the real world. The shift from real objects to models that can be implement in computers and used to study some problems, requires the development of special techniques. The study of these techniques is carried out by system analysis, which studies the structure of real objects and provides methods for their formalized description (virtualizations). *General systems theory* is a part of system analysis, which studies the nature of a variety of systems with the same approach. Systems analysis takes

a boundary position between theoretical informatics and cybernetics. The same boundary position is occupied by two more disciplines. *Simulation* is one of them; this science develops and uses special techniques for the reproduction of real processes on computers. The second science is *queuing theory,* which is the mathematical study of a special, but very broad class of models of information transmission and information processing, the so-called queuing system. Generally, models of queuing systems are constructed to predict queue lengths and waiting times.

5. *The Theory of Games and Special Behavior*. The last class of disciplines included in theoretical informatics focuses on the use of information for decision-making in a variety of situations encountered in the world. It primarily includes *decision theory*, which studies the general scheme used by people when choosing their solutions from a variety of alternatives. Such a choice is often the case in situations of conflict or confrontation; that is why models of this type are studied by *game theory*. A decision maker always wants to choose the best of all possible solutions. Problems which arise in a choice situation are studied in the discipline known as *mathematical optimization* also known as *mathematical programming*. To achieve goals, the decision making process must obey a single plan; the study of ways of building and using such plans is provided by another scientific discipline – *operations research*. If not individual, but team decision are made, there are many specific situations, e.g., the emergence of parties, coalitions, agreements and compromises. The problem of collective decision-making is examined in the *theory of collective behavior*.

#### 1.1.4.2 Applied Informatics

Cybernetics, which has already been mentioned, can be seen as the first applied informatics discipline concerned with the development and use of automatic control systems of diverse complexity. Cybernetics originated in the late 40s, when Norbert Wiener first put forward the idea that the control system in living and non-living artificial systems share many similarities. The discovery of this analogy promised the foundation of a general theory of control, the models of which could be used in newly-designed systems. This hypothesis has not withstood the test of time, but principles concerning information management systems have greatly benefited. *Technical cybernetics* was the most fully developed. It includes *automatic control theory*, which became the theoretical foundation of automation.

As the second applied informatics discipline, *programming* relies completely on computers for its appearance. In the initial period of its development, programming lacked a strong theoretical base and resembled the work of the best craftsmen. With experience, programming has groped towards general ideas that underlie the construction of computer programs and programming arrangements themselves. This has resulted in the gradual establishment of theoretical programming, which now consists of multiple destinations. One of them is connected with the creation of a variety of programming languages designed to make human interaction with computers and information systems easy.

*Artificial intelligence* is the youngest discipline of applied informatics, but now it determines the strategic directions of the development of information sciences. Artificial intelligence is closely related to theoretical computer science, from which it has borrowed many models and methods, such as the active use of logical tools to convert knowledge. Equally strong is its relation to cybernetics. The main objective of work in the field of artificial intelligence is the desire to penetrate the secrets of creative activity of people, their ability to master skills, knowledge and abilities. To do this, you need to open those fundamental mechanisms by which a person is able to learn almost any kind of activity. Such a goal of researchers in the field of artificial intelligence closely associates them with the achievements of psychology. In psychology, there is now a new area actively developing – *cognitive psychology*, which focuses exactly on examining the laws and mechanisms that interest specialists in the field of artificial intelligence. The sphere of interests of experts in the field of artificial intelligence also includes linguistic studies. Mathematical and applied *linguistics* also work closely with research in the field of artificial systems designed for natural language communication. Artificial intelligence is not a purely theoretical science; it touches on the applied issues related to the construction of real existing intelligent systems, such as robots.

The significant practical importance of informatics manifests itself in the field of *information systems*. This trend was created by researchers in the field of *documentology* (the scientific study of documents, including the examination of their structure) and by the analysis of scientific and technical information that were conducted even before computers. However, true success of information systems was reached only when computers become part of their composition. Now within this area, a few basic problems are being solved. The analysis and forecasting of various information streams, the study of methods of information presentation and storage, the construction of procedures to automate the process of extracting information from documents, and the creation of information search systems. On the one hand, research in the field of information systems is based on applied linguistics, which creates languages for the operative saving of information and for quickly finding answers to incoming requests in data warehouses. On the other hand, the theory of information supplies this research by models and methods that are used to organize the circulation of information in data channels.

*Computer engineering* is a completely independent line of applied research that integrates several fields of electrical engineering and computer science required to develop computer hardware. Within this field many problems which are not directly related to informatics are solved. For example, numerous studies are conducted to improve the element base of computers. The progress of modern informatics is unthinkable without the evolution of computers – the main and the only tool for working with various information. Research in the area of artificial intelligence has had great influence on the development of new computers. As a result, new generations of computers are far more intelligent than their ancestors. There are two major specialties in computer engineering – computer software engineers develop software, computer hardware engineers develop computer equipment. In the same way, computer engineers specialize in narrow sectors, e.g. coding and information protection, operating systems, computer networks and distributed systems, architecture of computer systems, embedded systems and so on.

The world is now entering the epoch of the information society. Certainly, the distribution, storage and processing of information will play a huge role in this society. Information becomes a commodity of great value, and information technology is an increasingly influential factor in business and in everyday life. The preparations for the transition of an information society are causing many problems, not only of a technical, but also of a social and legal nature. All of these problems are a subject of study for psychologists, sociologists, philosophers and lawyers who work in the field of informatics. Already, automated training systems, workstations for various specialists, distributed banking systems and many other information systems, whose operations are based on the full gamut of informatics have been created. The information technology sector, which contains these activities, can be defined as a *social informatics*.

There is one other area, in which informatics has recently played the role of an important accelerator of the research processes, i.e. natural sciences like biology, chemistry or physics. The main objective of this trend is the study of information processes in natural systems, and the use of acquired knowledge for the organization and management of natural systems and for the creation of new technical systems. *Informatics in the natural sciences* has its own characteristics, depending on the natural discipline.

## 1.2 Relationship with Some Fundamental Sciences

In the quite short historical period of its development, informatics has become a foundation for and taken inspiration from other fundamental sciences. At the same time, informatics has giving advances and inspirations to other disciplines. Sometimes, informatics has even merged with the name of a discipline, as in bioinformatics, health informatics or medical informatics; it denotes the specialization of informatics to the data management and information processing in the named discipline. The amalgamation of informatic theories and methods with classical (fundamental) disciplines enriches them too. Let us see how these interactions work.

### 1.2.1 Informatics and Mathematics

The mathematics of current computer science is constructed entirely on *discrete mathematics*, especially on *combinatorics* and on *graph theory*. Discrete mathematics (in contrast with continuous mathematics) is a collective name for all branches of mathematics, which deal with the study of discrete structures, such as graphs or statements in logic. Discrete structures contain no more than countable sets of elements. In other words, discrete mathematics deals with objects that can assume only separated values. Besides combinatorics and graph theory, these branches include cryptography, game theory, linear programming, mathematical logic, matroid theory, and number theory, which are used by informaticians intensively. The advantage is that nontrivial real world problems can be quickly explored by using methods of these discrete disciplines. One can even say that discrete mathematics constitutes the mathematical language of informatics.

From the beginning, informatics has been very solidly based in mathematics (Figure 1). In addition, theoretical informatics could be considered a branch of mathematics. It is easy to notice that both share a conceptual apparatus. The more precisely circumscribed area of computer science definitely includes many things, which would not be considered mathematics, such as programming languages or computer architecture. The computerization of sciences, including mathematics, also stimulated those sciences as well. Questions from informatics inspired a great deal of interest in some mathematical branches, e.g. in discrete math, especially in combinatorics. Some mathematical challenges arise from problems in informatics (e.g. complexity theory); they demand the use of innovative techniques in other branches of math (e.g. topology). Fundamental problems of theoretical computer science, like the P versus NP problem, have obtained an appropriate importance as central problems of mathematics.

The two-way conversation between informaticians and mathematicians is profitable. Mathematicians consider computational aspects of their areas to facilitate construction of the appropriate virtual objects (or entities, in terms of software). Numerous techniques increase mathematical sophistication, which result in efficiently solving the most important computational problems. Informatics has activated many mathematical research areas like computational number theory, computational algebra and computational group theory. Furthermore, a diversity of computational models was drafted to explain and sometimes anticipate existing computer systems, or to develop new ones like online algorithms and competitive analysis, or parallel and distributed programming models. Here are some explanations for the exemplary models:

– An online algorithm has a special restriction; it does not receive its input data from the beginning as a whole, but in batches (rounds). After each round, the algorithm has to provide a partial answer. An example is an allocation of CPU time or memory (scheduling), because in general it is not known which processes will

require resources, it is necessary to allocate resources only based on the current situation. Competitive analysis compares the performance of such algorithms to the performance of an optimal offline algorithm that can view the sequence of requests in advance.
– Distributed computation is a solution for the big data problem. Unfortunately, this is very difficult to program due to the many processes which are involved, like sending data to nodes, coordinating among nodes, recovering from node failure, optimizing for locality, debugging and so on. The MapReduce programming model is suggested which allows such data sets to be processed with a parallel, distributed algorithm on a computer cluster. The model was inspired by functional programming. Applications which have implement the MapReduce framework, achieve high scalability and fault-tolerance, which is obtained by optimizing the execution engine.

### 1.2.2 Informatics and Mathematical Logic

*Mathematical logic* or *symbolic logic* is a branch of mathematics that studies mathematical notation, formal systems, verifiable mathematical judgments, computability, and the nature of mathematical proof in general. More broadly, mathematical logic is regarded as a mathematized branch of formal logic or logic, developed with the help of mathematical methods. Topically, mathematical logic stands in close connection with meta-mathematics, the foundations of mathematics, and theoretical computer science. Logic in computer science is the direction of research, according to which logic is applied in computation technologies and artificial intelligence. Logic is very effective in these areas. However, one should not forget that some important research in logic were caused by the development of computer science, for example, applicative programming, computation theory and computational modeling.

From the very beginning, informatics depends heavily on logic; do not forget that Boolean logic and algebra was used for the development of computer hardware. Logic is a part of information technology, for example, in relational data models, relational databases, relational algebra, and relational calculus. In addition, logic has provided fundamental concepts and ideas for informatics, which naturally can use formal logic. For example, this applies to the semantics of programming languages. Here are some very important applications of logic in the field of informatics:
– Formal methods and logic reasoning about concepts in semantic networks and semantic web;
– Problem solving and structured programming for application development and the creation of complex software systems;
– Probative programming – the technology of development of algorithms and programs with proof of the correctness of algorithms;

- The logic of knowledge and justified assumptions, e.g. in artificial intelligence;
- Prolog language and logic programming for creating of knowledge bases and expert systems and research in the field of artificial intelligence;
- Spatial and temporal logic to describe spatial position and movement;
- Abstract machine logic, especially code compilation and optimization;
- Objects transformation based on lambda calculus.

### 1.2.3 Informatics and Cybernetics

Cybernetics investigates abstract principles of organization of complex systems [7] (not only artificial, but also natural systems). It deals with the functioning of the systems, rather than their design. The fundamental contribution of cybernetics was its explanation of purposiveness of natural systems using key concepts of *information* and *control*. Cybernetics focuses on the use of information, models and the control actions by systems to achieve their goals and to prevent disorder at the same time. The historical relationship between computer science and cybernetics has already been mentioned in section 1.1. In the XX century, cybernetics was presented as one of the main areas of computer technology; theoretical informatics was named mathematical cybernetics; cybernetics was named the first applied informatics discipline.

Two disciplines born from cybernetics, computer science (informatics) and control engineering had become fully independent in the early 1970s. Informatics has taken over the terminology (conceptual base) of cybernetics in the areas of equipment control and information analysis. Some areas of modern informatics have their origins directly in cybernetics. The best-known areas are information theory, robotics, the study of systems, and the theory of computing machines. After this, the remaining cyberneticist have differentiated themselves from mechanistic approaches by emphasizing the phenomena of autonomy, self-organization, cognition, and the role of the observer in modeling a system. This movement became known as second-order cybernetics. Nowadays, cybernetics is not such a popular scientific discipline; modern theories have divided its topics.

### 1.2.4 Informatics and Electronics

Transistors are the smallest active electrical components of modern computers. They operate as elements of integrated circuits. For example, microprocessors are physically integrated circuits, which are produced for computers in the form of chips. Consisting of transistors, chips can perform arithmetical and logical operations, store data and communicate with other chips. Electronic computer circuits operate in digital mode; this means that all the transistors of the integrated circuits of computers can be only in specific states. Popular digital electronic circuits operate in binary mode, so these specific states are only two; they are referred to as *zero* and *one*. The theory of design and operation of the various electronic components like transistors is carried out by *electronics*, which is a part of *electrical engineering* and a branch of physics. To emphasize the type of basic materials used in modern electronics components and circuits, the terms *semiconductor electronics* or *solid state electronics* are used. To underline the digital operating mode of electronic circuits intended for the construction of computers, the term *digital electronics* is used.

*Computer engineering* fuses electronics with computer science to develop faster, smaller, cheaper, and smarter computing systems. Computer engineers design and implement hardware (computer architectures); design and implement systems software (operating systems and utility software); design processors; design and implement computer-computer and human-computer interface systems. One can say that computer engineers are engaged in analyzing and solving computer-oriented problems. Computer engineering also deals with the embedded computers that are parts of other machines and systems, as well as with computer networks, which are developed for data transfer.

It is an interesting fact that progress in micro-miniaturization of integrated electronic circuits has been and remains heavily supported by informatics. There have long been popular industrial software tools for designing electronic systems such as printed circuit boards and integrated circuits called ECAD (electronic design automation).

### 1.2.5 Informatics and Linguistics

*Computational linguistics* joins linguistics and informatics; it is one of the cognitive sciences, which partly overlaps with the field of artificial intelligence and concerns understanding natural language from a computational perspective. Theoretical computational linguistics deals with formal theories about language generation and understanding; the high complexity of these theories forces the computer management of them. One of the main goals of theoretical study [8] is the formulation of grammatical and semantic frameworks for characterizing languages enabling a computational approach to syntactic and semantic analysis. In addition, computational linguists discover processing techniques and learning principles that exploit both the structural and distributional properties of language. This kind of research is not possible without the answer to the question: how does language learning and processing work in the brain?

Applied computational linguistics, called *language engineering*, focuses on the methods, techniques, tools and applications in this area. Historically, computational linguistics was initiated to solve the practical problem of automatic written text translation. Machine translation was recognized as being far more difficult than had originally been assumed. In recent years, we can observe the growth of technology for analysis and synthesis of spoken language (i.e., speech understanding and speech generation). Besides designing interfaces to operate in a natural language, modern computational linguistics has considerable achievements in the field of document processing, information retrieval, and grammar and style checking.

Computational linguistics partially overlaps with *natural language processing*, which covers application methods of language description and language processing for computer systems generally. This entails:

- Creation of electronic text corpora;
- Creation of electronic dictionaries, thesauri, ontologies;
- Automatic translation of texts;
- Automatic extraction of facts from texts;
- Automatic text summarization;
- Creation of natural language question-answering systems;
- Information retrieval.

Computational linguistics also embraces topics of computer-assisted second language learning. It is not only about the use of computers as an aid in the preparation, presentation, and assessment of material to be learned. The modernization of educational methods, such as the utilization of multimedia, and web-based computer assisted learning is also sought.

### 1.2.6 Informatics vis-à-vis Psychology and Sociology

There are more reasons for social, behavioral, or cognitive scientists (psychologists and sociologists) to acquire a basic familiarity with informatics tools and techniques, which give them a new ability to provide, publish and evaluate their research. One can speak about a new phenomenon – computational social science [9]. Here is an incomplete list of what a new computational approach can bring to psychology and sociology:

- Web-based data collection methods;
- Mobile data collection method;
- Data manipulation and text mining;
- Computerized exploratory data analysis and visualization;
- Big Data and machine learning applications.

A new, widely unknown discipline *psychoinformatics* [10] already helps psychologists to mine data and to develop patterns based on relations among data; the pattern finally reflects specific psychological traits. Furthermore, the development of psychology and informatics is on such a stage, when psychology can try to model the information processes of human thought. Some cognitive scientists even define metaphorically psychology as the informatics of the human mind [11]. In their view, the self-organization structures in the brain leads to the arrangement of stable data objects (images), most of which have prototypes in the external world or in the human body. The aggregate of objects forms a working model – a personal image of the world. Supposedly, primary information objects are formed using the innate transformation of sensory signals; the same transformation can be applied recursively to the primal images, creating images in the next row that do not have any direct prototypes in the outside world, nor in the human body.

At the same time, informatics, especially artificial intelligence, can be used with a positive effect as psychological models of human behavior. For example, some mental problem solving techniques studied in psychology are used as prototypes in artificial intelligence. Artificial intelligence is implemented as a computer or other device that has been designed "to think like a human". Software engineers also use psychological knowledge, especially to design human-computer interactions; there is a separate presentation layer of software, which is designed by specialists in graphical user interfaces. Sociological knowledge is very useful in the context of managing large IT project teams. Separately, we can note the existence of *social informatics* – an interdisciplinary research field that studies the information society.

### 1.2.7 Informatics and Economics

Intensive implementation of information technology in the economy has led to a new area in computer science called *business informatics*. Business informatics (from the German *Wirtschaftsinformatik* or organizational informatics) is an integrated applied discipline, based on the interdisciplinary links between informatics, economics, and mathematics. Economic information is the aggregate data reflecting the status and progress of business processes. Economic information circulates in the economic system and the accompanying processes of production, distribution, exchange, and consumption of goods and services. It should be considered as one of the types of management information. Business informatics deals with knowledge about information systems, which are used for preparation and decision-making in management, economics and business. This discipline includes information technology and significant elements oriented on construction and implementation of information systems. In contrast to information systems theory, business informatics focuses on solutions, not on empirically explaining phenomena of the real world.

Economics is a stable source of tasks for computer engineers; e.g., the stock market delivers huge data sets for computer analysis and prediction of events. At the same time, economics inspires theoretical computer science, because business needs more and more new algorithms, feeling the heat of competition. Historically, economic initiated one of the first grand accomplishments in the algorithm theory – the simplex method, which has proved to be very effective in decision-making problems.

The impact of the economy on IT can be seen in the formation of an agent-based programming paradigm. The idea for this paradigm came from a commercial brokerage model. A software agent is a kind of small independent program, which mediates the exchange of information between other programs or people. Agents must be prepared to receive incorrect data from another agent, or to receive no data at all; this means that agent-based systems should be prepared to function in conditions of uncertainty. Agent-based modeling promises new analytical possibilities not only for business, but also for social sciences. At present, computing scientists expect help from economists to solve the problems of optimization of software agents.

Completely new types of algorithmic problems from natural sciences are challenging theoretical informatics: namely, problems in which the required output is not well defined in advance. Typical data might be a picture, a sonogram, readings from the Hubble Space Telescope, stock-market share values, DNA sequences, neuron recordings of animals reacting to stimuli, or any other sample of "natural phenomena". The algorithm (like the scientist), is "trying to make sense" of the data, "explain it", "predict future values of it", etc. The models, problems and algorithms here fall into the research area of computational learning and its extensions.

## 1.3 Information Theory

The term information entered into scientific use long before the rapid development of electronic communications and computing. Information (from the Latin word *informatio*, which means clarification, presentation, interpretation) in the most common sense means details about something transmitted by people directly or indirectly. Initially, this concept was only associated with communicative activities in the community. The understanding of information to be messages sent by people (oral, written or otherwise, like conventional signals, technical facilities, etc.), remained until the mid 20-ies of XX century.

Gradually, the concept of information has gained a more and more universal meaning. Before the beginning of the 1920s, the information was treated on a qualitative level, any formal concepts, procedures and methods of quantifying were not used. The main focus was on the mechanisms of influence on receivers of information, to the ways of insurance its accuracy, completeness, adequacy, etc. The subsequent refinement of information's scientific meaning was done by scientists in different directions. At first, they tried to include it as part of the structures of other general concepts (e.g. probability, entropy, and diversity). R. S. Ingarden (1963) proved the failure of these attempts.

The development of the technical facilities of mass communication (the telephone, telegraph, radio, television, computer networks) led to a snowballing in the number of transmitted messages. This led to the need to evaluate various characteristics of information, and in particular its volume. With the development of cybernetics, information become one of the main categories of its conceptual apparatus, along with concepts such as management and communication.

Nowadays, the field of information theory partly belongs to mathematics, computer science, physics, electronics and electrical engineering. Important subfields of information theory are measures of information, source and channel coding, algorithmic complexity theory, information-theoretic security, and others. Information theory investigates the processes of storage, conversion and transmission of information. All this processes are based on a certain way of measuring the amount of information. A key measure of information is entropy, which is expressed by the average number of bits needed to store or communicate one symbol in a message. Entropy quantifies the uncertainty involved in predicting the value of a signal.

As it was first applied, mathematical information theory was developed by Claude E. Shannon (1948) to find fundamental limits on signal processing operations. It was based on probabilistic concepts about the nature of information. The area of the correct use of this theory is limited to three postulates: 1) only the transmission of information between technical systems is studied; 2) a message is treated as a limited set of characters; 3) the semantics of messages is excluded from the analysis. This theory gave engineers of data transmission systems the possibility of determining the capacity of the communication channels. It focuses on how to transmit data most efficiently and economically, and how to detect errors in its transmission and reception. Information theory is sometimes seen as the mathematical theory of information transmission systems, because the primary problems of this science arose from the domain of communication. It establishes the basic boundaries of data transmission systems, sets the basic principles of their development and practical implementation. Since its inception information theory has broadened to find applications into many other areas, e.g. data compression and channel coding.

Shannon's theory, which studies the transmission of information, is not engaged in the meaning (semantics) of the transmitted message. As an attempt to overcome its limitations, new versions of Shannon's mathematical theory of information appeared: topological, combinatorial, dynamic, algorithmic and others. However, they take into account only the symbolic structure of messages, and they can be attributed only to theories of syntactic, not semantic type. There is a later, complementary piece of information theory that draws attention to the content through the phenomenon of lossy compression of the subject of the message, using the criterion of accuracy. Unfortunately, the attempt made to connect a probabilistic-statistical (syntax) approach and semantic approach has not led yet to any constructive results. Nevertheless, it may be noted that the category information has infiltrated the relevant scientific fields, so it has obtained the status of a general scientific concept. To the original meaning of information were added:

– A measure of reducing the uncertainty as a result of the message;
– A collection of factual knowledge, e.g. circulating in the management process;

– The process of meaningful reflection of the diversity of natural objects, exchange of signals in the animal and plant world, the transmission of traits from cell a cell;
– The communication not only between machines, but also between man and machine, and between people.

### 1.3.1 Quantities of Information

Information is always presented as a message. The elementary units of messages are characters; characters assembled in a group create words. A message is issued in the form of words or characters is always transmitted in a material and energy form (e.g. an electric or light channel). When using structural measures of information only the discrete structure of the message, the amount of information elements it contains, the connections between them are taken into account. In a structural approach, geometric, combinatorial and additive measures of information are distinguished.
Geometric measure involves the measurement of the parameter of the geometrical model of the message (length, area, volume) in discrete units. A combinatorial measure of the amount of information is defined as the number of combinations of elements (characters). The possible amount of information coincides with the number of possible combinations, permutations and placement of elements. The most common is the additive measure (Hartley's measure) in accordance with it the amount of information is measured in binary units - bits.

#### 1.3.1.1 Units for Measuring Computer Information

In computer technologies, the amount of information is the capacity of a data storage system or memory. Information capacity refers to a count of binary symbols stored in the memory; that is why it is a dimensionless quantity. In information theory, the units of information are used to measure the information contents (entropy) of sources of information and of communication channels. In both areas, the most common units of information are the bit and the byte (one byte is equivalent to eight bits). One-bit storage can save only two different symbols – 1 or 0 (logical *yes* or *no*). One-byte storage can save 256 different symbols – 0, 1, 2, 3...255. This means that a data storage system with 256 possible states can store up to $\log_2 256 = 8$ bits or 1 byte of information. The first, Ralph Hartley (1928), observed the logarithmic dependence of the amount of information that can be stored in a system on possible states of that system.
Multiples of information units are formed from bits (b) and bytes (B) with the SI[1]

1 International System of Units

or IEC[2] prefixes (see table 1). Units of information are not covered formally in the International System of Units, but computer professionals have historically used the same symbols and pronunciation for the binary series in the description of computer storage and memory. The simple reason was: *1 kB = 1024 B ≈ 1000 B*, 1MB = 1024 kB ≈ 1000 kB, but now a lot more information capacity is used and rounding errors become more noticeable, hence the necessity of the use of binary IEC units.

**Table 1:** Decimal and binary prefixes of information units.

| Decimal (SI) | | | Binary (IEC) | | |
|---|---|---|---|---|---|
| Value | Symbol | Name[3] | Value | Symbol | Name |
| 1000 | *kB* | kilobyte | 1024 | *KiB* | kibibyte |
| $1000^2$ | *MB* | megabyte | $1024^2$ | *MiB* | mebibyte |
| $1000^3$ | *GB* | gigabyte | $1024^3$ | *GiB* | gibibyte |
| $1000^4$ | *TB* | terabyte | $1024^4$ | *TiB* | tebibyte |
| $1000^5$ | *PB* | petabyte | $1024^5$ | *PiB* | pebibyte |
| $1000^6$ | *EB* | exabyte | $1024^6$ | *EiB* | exibyte |

### 1.3.1.2 Quantities of Information in Information Theory

| | |
|---|---|
| **Q&A**<br><br>“What is information?” | **Technical definition:** Information is stimuli that have meaning in some context for its receiver. In computer science, information is a choice or entropy. The meaning of a message (in the human sense) is irrelevant. When information is entered into computer through interfaces and is stored in a computer memory, it is generally referred to as data. After processing (e.g. formatting and printing), output data can again be perceived as information.<br>**Business definition:** Information is data that is accurate and timely, specific and organized for a purpose, presented within a context that gives it meaning and relevance, and can lead to an increase in understanding and decrease in uncertainty. Information is valuable because it can affect behavior, a decision, or an outcome. A piece of information is considered valueless if, after receiving it, things remain unchanged. For a technical definition of information, see information theory. http://www.businessdictionary.com |

2 International Electrotechnical Commission

3 In the microelectronics industry, the names kilobyte (KB), megabyte (MB), and gigabyte (GB) are used for the value 1024 according to JEDEC standard, https://www.jedec.org/.

In information theory, the concept of information has been introduced as axiomatic (without explanation). Instead of answering the question „What is the information?“, the founders of information theory answer the question „How can we measure information?“ Harry Nyquist and Ralph Hartley (research leaders at Bell Labs when Shannon arrived in the early 1940s) were the most direct precursors of Shannon‘s theory. In considering the engineering of telegraph systems, Nyquist (1924) discusses *quantifying intelligence* – information transmitted by telegraph signals – and the highest *line speed* at which signals can be transmitted by a communication system. He gives the relation

$$W=K\log(m), \tag{1}$$

where $W$ is the speed of transmission of intelligence, $m$ is the number of different voltage levels to choose from at each time step, and $K$ is a constant.
Hartley (1928) first used the term *transmission of information* in a technical sense, and makes explicitly clear, that information in this context was a measurable quantity. The amount of information he understood as the receiver’s ability to distinguish that the specific sequence of symbols had been intended by the sender rather than any other, and quantified as

$$H=\log(S^l), \tag{2}$$

where $S$ was the number of possible symbols in the communication channel, and $l$ – the length of message or in other words the number of symbols in the transmission. Intuitively, this definition makes sense: one symbol (e.g. one letter) has the information of $\log(S)$, then a sentence of length $l$ should have $l$ times more information.

The natural unit of information was therefore the decimal digit, much later renamed the hartley in his honor as a unit or scale or measure of information. The logarithm of total number of possibilities $H0$ is still used as a quantity for the Hartley information. This parameter characterizes the theoretical capacity of the information channel.

Shannon’s measure of information is the number of bits to represent the amount of uncertainty (randomness) in a data source, and is defined as entropy

$$H=-\sum_{i=1}^{n} p_i \log(p_i)\,. \tag{3}$$

Where there are $n$ symbols *1, 2 …n*, each with probability of occurrence of $p_i$.

Once again, it is worth recalling that Shannon was building a mathematical theory of communication. The amount of information in a Shannon information channel – is the number of random data at one site relative to another. Let $x$ and $y$ – the random variables defined on the corresponding sets $X$ and $Y$. Then the amount of information $x$ of $y$ is the difference between a priori and a posteriori entropy:

$$I(x,y)=H(x)\text{-}H(x|y), \tag{4}$$

where

$$H(x)=\sum_{x}^{x\ in\ X} p(x)\log_2 p(x), \tag{5}$$

$$H(x\,|\,y)=-\sum_{y}^{y\ in\ Y} p(y)\sum_{y}^{y\ in\ X} p(x\,|\,y)\log_2 p(x\,|\,y). \tag{6}$$

Here, equation (5) describes the entropy. Equation (6) describes the conditional entropy, in communication theory it characterizes the noise in the communication channel.

### 1.3.2 Coding Theory

Coding theory is the study of the properties of codes and their fitness for a specific application. Codes are used for data compression, cryptography, error-correction and more recently also for network coding. Codes are studied by various scientific disciplines—such as information theory, electrical engineering, mathematics, and computer science—for the purpose of designing efficient and reliable data transmission methods. This typically involves the removal of redundancy and the correction (or detection) of errors in the transmitted data. There are essentially two aspects to coding theory:

1. Data compression (source coding)
2. Error correction (channel coding)

Source encoding attempts to compress the source data in order to transmit them more efficiently. This practice is found every day on the Internet where Zip data compression is used to reduce the network load and make files smaller.

Channel encoding, adds extra data bits to make the transmission of data more robust to disturbances present on the transmission channel. The ordinary user may not be aware of many applications using channel coding. A typical music CD uses the Reed-Solomon code to correct for scratches and dust. In this application, the transmission channel is the CD itself. Cell phones also use coding techniques to correct for the fading and noise of high frequency radio transmission. Data modems, telephone transmissions, and NASA all employ channel-coding techniques to get the bits through, for example the turbo code and LDPC codes.

Coding theory is a branch of applied mathematics concerned with transmitting data across noisy channels and recovering the original message. Coding theory is about making messages easy to read. Please, do not confuse it with cryptography which is the art of making messages hard to read.

Typically, the message in communication channels (such as a telephone line or CD stream) is in the form of binary digits or bits, strings of 0 or 1. The transmission of these bits takes place along a channel in which errors occur randomly, but at a predictable overall rate. To compensate for the errors we need to transmit more bits than there are in the original message.

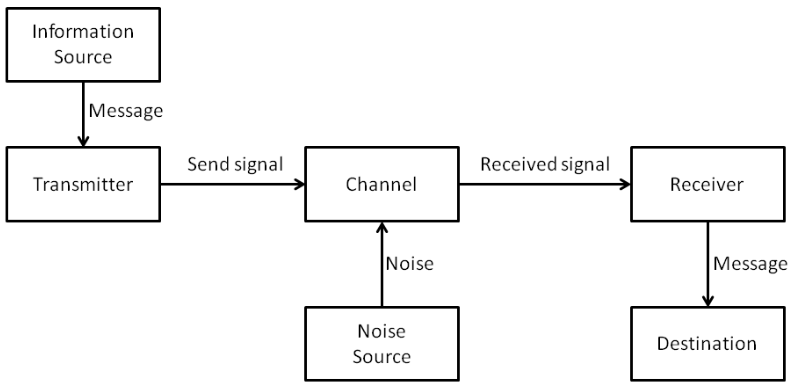


**Figure 2:** Shannon's model of communication.

The simplest method for detecting errors in binary data is the parity code, which transmits an extra "parity" bit after every 7 bits from the source message. However, this method can only detect errors; the only way to correct them is to ask for the data to be transmitted again.

A simple way to correct as well as detect errors is to repeat each bit a set number of times. The recipient sees which value, 0 or 1, occurs more often and assumes that that was the intended bit. The scheme can tolerate error rates up to one error in every two bits transmitted at the expense of increasing the amount of repetition.

### 1.3.3 Semiotics

Semiotics is the study of signs and symbols and their use or interpretation. It is the study of meaning making, the philosophical theory of signs and symbols. This includes the study of signs and sign processes (semiosis), indication, designation, likeness, analogy, metaphor, symbolism, signification, and communication. Semiotics is closely related to the field of linguistics, which, for its part, studies the structure and meaning of language more specifically. Where it differs from linguistics, however, is that semiotics also studies non-linguistic sign systems. Semiotics often is divided into three branches:

1. Semantics is the study of meaning, i.e. of the relationship of signs to what they stand for the relation between signs and the things (denotata) to which they refer.

2. Syntactics is the study of relations among signs in formal structures. Syntactical knowledge goes farther than what is grammatical; it also studies ambiguity, in which a sentence could have more than one meaning, and enables us to determine grammatical categories such as subject and direct object. It enables the use in sentences of terms, which we may not consciously be able to define.
3. Pragmatics is the study of the relation of signs to interpreters' sign-using agents; it is the interpretation of linguistic meaning in a given context. There can be a difference between linguistic context (the discourse that precedes a sentence to be interpreted) and situational context (the knowledge about the world). That is why comprehension the meaning of sentences can be a difficult task.

Semiotics is the science of communication and sign systems as well. In short, it is the science of the ways in which people understand phenomena and organize them mentally. It is also the science of the ways, in which people devise means for transmitting that understanding and for sharing it with others. Although natural and artificial languages are therefore central to semiotics, the field covers all non-verbal signaling and extends to domains whose communicative dimension is perceived only unconsciously or subliminally. Knowledge, meaning, intention and action are thus fundamental concepts in the semiotic investigation of phenomena.

Semiotics is often employed in the analysis of texts, which can exist in any medium and may be non-verbal. The term *text* usually refers to a message, which has been recorded in some way (e.g. writing, audio- and video-recording) so that it is physically independent of its sender or receiver. A text is an assemblage of signs (such as words, images, sounds, and gestures) constructed and interpreted with reference to the conventions associated with a genre and in a particular medium of communication.

The term *medium* is used in a variety of ways by different theorists, and may include such broad categories as speech and writing or print and broadcasting or relate to specific technical forms within the mass media (radio, television, newspapers, magazines, books, photographs, films and records) or the media of interpersonal communication (telephone, letter, fax, e-mail, video-conferencing, computer-based chat systems). Some theorists classify media according to the communication channels involved visual, auditory, tactile and so on. Human experience is inherently multisensory, and every representation of experience is subject to the constraints and allowances of the medium involved. Every medium is constrained by the channels, which it utilizes. Different media and genres provide different frameworks for representing experience, facilitating some forms of expression and inhibiting others. The more frequently and fluently a medium is used, the more transparent (invisible) to its users it tends to become.

Semiotics represents a range of studies in art, literature, anthropology and the mass media rather than an independent academic discipline. Those involved in semiotics include linguists, philosophers, psychologists, sociologists, anthropologists, literary, aesthetic and mass media theorists, psychoanalysts and educationalists. Beyond the most basic definition, there is considerable variation amongst leading semioticians as to what semiotics involves. It is not only concerned with (intentional) communication but also with our assignment of significance to anything in the world.

Semiotics teaches us not to take reality for granted as something having a purely objective existence, which is independent of human interpretation. It teaches us that reality is a system of signs. Studying semiotics can assist us to become more aware of reality as a construction and of the roles played by others and ourselves in constructing it. It can help us to realize that information or meaning is not contained in the world or in books, computers or audio-visual media. Meaning is not transmitted to us. We actively create it according to a complex interplay of codes or conventions of which we are normally unaware. According to semiotics, we live in a world of signs and we have no way of understanding anything except through signs and the codes into which they are organized.

#### 1.3.3.1 Computational Semiotics

*Computational semiotics* is an interdisciplinary field that applies, conducts, and draws on research in logic, mathematics, the theory and practice of computation, formal and natural language studies, the cognitive sciences generally, and semiotics proper. A common theme of this work is the adoption of a sign-theoretic perspective on issues of artificial intelligence and knowledge representation. Many of its applications lay in the field of human-computer interaction (HCI) and fundamental devices of recognition.

Computational Semiotics refers to the attempt of emulating the semiosis cycle within a computer. It refers to the study of the computer as a medium or distinct sign system and its central concern is the extent to which the information inside a computer can be considered semiotic or can be usefully analyzed in semiotic terms. Among other things, this is done for the purpose of constructing autonomous intelligent systems able to perform intelligent behavior, what includes perception, world modeling, value judgment and behavior generation. There is a claim that most intelligent behavior should be due to semiotic processing within autonomous systems, in the sense that an intelligent system should be comparable to a semiotic system. Mathematically modeling such semiotic systems is being currently the target for a group of researchers studying the interactions encountered between semiotics and intelligent systems.

The key issue in this study is the discovery of elementary or minimum units of intelligence, and their relation to semiotics. Some attempts have been made to determine such elementary units of intelligence, i.e., a minimum set of operators that would be responsible for building intelligent behavior within intelligent systems. Within computational semiotics, we try to depict the basic elements composing an intelligent system, in terms of its semiotic understanding. We do this by the definition of a knowledge unit, from which we derive a whole taxonomy of knowledge. Different types of knowledge units are mathematically described and used as atomic components for an intelligent system. They are at the same time, containers of information and active agents in the processing of such information.

Computational semiotics is a branch of one, which deals with the study and application of logic, from computation to formal, to natural language in terms of cognition and signs. One part of this field, known as algebraic semiotics, combines aspects of algebraic specification and social semiotics, and has been applied to user interface design and to the representation of mathematical proofs.

Semiotics deals with the basic ingredients of intelligence [12] (signs or representations, object or phenomenon, interpretants or knowledge) and their relationships, the triple (sign, object, interpretant) represents a signic process. To perform semiosis is to extract meaning of an object or phenomenon. In the process of extracting meaning, it essentially studies the basic aspects of cognition and communication. Cognition means to deal with and to comprehend phenomena that occur in an environment. Communication means how a comprehended phenomenon can be transmitted between intelligent beings.

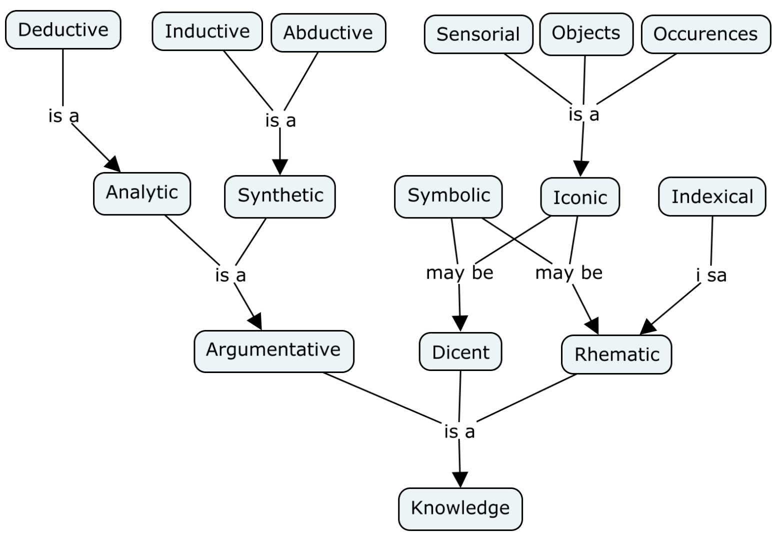


**Figure 3:** The map[4] of knowledge unit concepts classified by its nature (in accordance with Ricardo R. Gudwin [13]).

4 The map of concept (concept map) includes concepts, enclosed in boxes with rounded corners, and relationships between concepts or propositions. Relationships are indicated by a connecting line and linking words, which specify the relationships between concepts.

Signs are the basic units of analysis in semiotics; they are representation, which are either internal or external to a cognitive system, which can evoke internal representations called interpretants that represent presumable objects in the world. Interpretants, too can act as signs evoking other interpretants. The taxonomy of signs provides three distinct kinds of knowledge, relates to the experience of an object, to experience of a sign of an object and to the other experience of the mapping between these two concepts. A sign can be a *mere quality*, an *actual existent* or a *general law*; a sign interpretant represents it as a sign of possibility (*rheme*), fact (*dicent*) or reason (*argument*).

Basic rhematic knowledge[5] can be:

- Symbolic – is knowledge of arbitrary names like 'dog' or any symbol which has a conventional and arbitrary relationship to its referent;
- Indexical – is knowledge of indices – signs that are not conventionalized in the way symbols are but are indicative of something in the way that smoke is indicative of fire;
- Iconic – is knowledge of signs that resemble their referents or provide direct models of phenomena (as such, icons unlike symbols are not arbitrarily related to their referents.

Dicent knowledge employs truth-values to link sensorial, object or occurrence knowledge to world entities. It has two types: iconic (truth-values of iconic proposition are derived directly from iconic rhematic knowledge) and symbolic (truth-values of symbolic proposition match those of their associated propositions, because symbolic propositions are names for other propositions, which may be either iconic or symbolic).

Argumentative knowledge is knowledge used to generate new knowledge through inference or reasoning. There are three types: deductive (is categorized as analytic, meaning it does require knowledge of the word), inductive (synthetic, involves inference from a large number of consistent examples and a lack of counter examples) and abductive (synthetic, valid inferences do not contradict previous facts).

5 Pertaining to the formation of words or to the rheme of a sentence.