Theory of Computation

Comments are appreciated!

Amir Daneshgar

First Version: February 26, 2020
Last Version: December 19, 2023

Amir Daneshgar

Sharif University of Technology
http://sharif.ir/~daneshgar
@ daneshgar@sharif.ir
✉ 11155-9415, Tehran, Iran
☎ (+98) 21 6616 5610

Contents

Chapter 1

On the concept of computation

MW![1]

[1] This exposition is clearly biased by the author's personal experience, views and limited knowledge.

1.1 Some historical notes

General intuitive comments.

Theory of computation (or ToC for short) is an adolescent, if not a child, and in the middle of its path toward machurity. The fact is, what we currently know as ToC is formed of different parts, each being developed in a separate discipline of science and technology. As a result, what we are facing today, although pasted together appropriately to present a well-defined theory, unfortunately does not poccess a unifying language or style, where a careful observer may still trace such inconsistencies every here and there throughout the theory. Although we are quite sure nowadays that we are standing on firm grounds as far as ToC is concerned, but such multilingualism is definitely a drawback when one is trying to teach or present the concepts. Hence, a crucial first step is to learn about the history of the subject and all different disciplines in which various parts of this theory have been developed. This, in its own right, not only helps to understand the nomenclature but also is quite interesting since one may find out that some very deep concepts of the theory actually were developed when nothing similar to what we call a *computer* nowadays was available at the time. What follows is an attempt to present such a consice historic view to the subject regardless of archaeological or historical technicalities.

Human brain has developed during the past 40000 years or more, to discover and invent many different and fascinating concepts, among which, most probably, *thinking*, *reasoning* and *computing* seem to be most fundamental and intriguing. Although, these concepts are almost as old as mankind's history, a formal and axiomatic approach, at least to reasoning and computing dates back not more than a century ago, within the realm of still developing field of *theoretical computer science*, while it seems quite out-of-reach of the theory, at the time being, to reconstruct our own brain, as a computational masterpiece.

This book can be considered as a starter to this rapidly growing field, concentrating on the concept of *computability*[2], and to begin, it is definitely of interest and quite necessary to have a fair idea about the history of the subject. In this section we provide a very concise and abridged history[3] of *computation* with an emphasize on turning points and development of the concepts related to what are going to be discussed later.

1.1.1 Ancient times

There are archaeological evidence, dating back to about 30000 years ago, that mankind have been able to *count* for quite a long time, however, it seems that it also took a long time too for human to master the concept, invent number systems and solve equations. It is not far from reality if one characterizes our computational achievements during ancient times (up to *circa* 500 CE) as the invention of *efficient number systems* on the one hand, and the *efficient use of geometry* to solve equations on the other. It is interesting to note that both concepts have gone through severe metamorphosis, during the middle ages and after that in modern times, a gist of which will be discussed later.

[2] Other topics of modern theoretical computer science, as the theory of *computational complexity*, are not our main concern in this book

[3] This section ought not to be considered as a technical historical note. Almost all referenced dates are approximations to indicate the chronological sequence of events unless we are sure about them as historical facts, while we have tried to avoid controversial historical issues or claims that are not unanimously accepted by experts. Above all, it goes without saying that, author's view has influenced the presentation and some scientific conclusions, of which he takes full responsibility. The interested reader is referred to the extensive existing literature as well as the historical and archaeological evidence (e.g. [20, 29, 53]), along with what are referenced within the text itself (comments and criticisms are highly appreciated).

Number systems as standards to write *numbers* using *digits* and the operations on them, on the one hand, and languages as standards to write *words* using *alphabets* and the concatenation of them making sentences, on the other hand, are similar in syntax while quite different in semantics and applications. As a matter of fact, our knowledge about formation of languages and number systems are not accurate at all and is mainly based on archaeological evidence. Both of these concepts first were invented orally and then at some time mankind was able to write and invented *symbols*. This, as a fundamental breakthrough, made our ancestors capable of recording what were happening in their brains, and *communicate* , in a more systematic way than painting, and then a long and interesting story started, of which, we possess some tiny pieces of evidence, as printouts of those ancient natural computing machines.

By historical evidence, we know that *circa* 2500 BCE, people of Sumer and Egyptians were able to write with symbols, and people of Sumer had a sexagesimal number system for calculation, later developed and enhanced by Babylonians, of which base 60 still appears in our clocks and measurement of angles in terms of degrees. A turning point of the history is the outstanding enhancement of seemingly primitive Sumerian alphabet and number system by the Babylonians *circa* 2300 BCE, while they were able to introduce the first positional cuneiform sexagesimal number system (*circa* 1800 BCE) that we are aware of[4].

It is interesting to note that, about the same time, Babylonians were also able to improve Sumerian *Abacus* as one of the oldest computational instruments we know of (see [21] for a Figure(s)), variants of which is still in use in some countries as China, Iran, Japan and Russia. We are also aware of some other ancient mechanical instruments which were used for some sort of computation, mainly motivated by astronomy and navigation[5], as *astrolabe* that was invented in the Hellenistic civilization by Apollonius of Perga *circa* 200 BCE, often attributed to Hipparchus (see [23] for a Figure(s)), which was used and considerably improved later by Persian astronomers. An interesting piece of machinery discovered in a Greek shipwreck off the island of Antikythera in 1901, is examined and estimated to date back to *circa* 100 BCE is called the *Antikythera mechanism* that seemingly had used to be used for some astronomical purposes (see [22] for a Figure(s)).

We know that at *circa* 1500 BCE the first alphabetic language that used *letters* was invented by Semites of Mesopotamia. Also, we know that apart from the old sexagesimal number systems already available, there were also number systems that used powers of ten, as the cuneiform number system used around 1500 BCE by Semites of Mesopotamia, or number systems that used multiplication and division by two, as what was used by Egyptians around 1700 BCE. It is interesting to note that the concept of *zero* was only known to Babylonians in their sexagesimal number systems since 300 BCE, while it was not recognized as a number in spite of the fact that there was a cuneiform character for it.

As we know that *binary number system*, containing zero as a number, was used by the Indian mathematician Pingala, *circa* 200 BCE, while positional *decimal number system*[6] was also seriously studied and applied by Indian mathematician and astronomer Brahmagupta (*circa* 598 CE – 668 CE). He used zero as a place holder and created decimal digits, improving the methods already introduced by the Indian astronomer Âryabhata, outsetting the age of modern number systems [52].

On the other hand, there are archaeological and historical evidence to show that people in Egypt and Mesopotamia used mathematical rules for measurement, surveying and architecture even at *circa* 3000 BCE. Also, we know from discovered clay tablets, that Babylonians were aware of many geometric constructions and measurements, e.g. the Pythagorean theorem, and used them for more that 1500 years until the fall of Babylon at 539 BCE. However, it was the Greek civilization, who around the same time (during the 6th century BCE), gathered and organized these methods to form the scientific field of study called *geometry*, where the name itself, as a combination of Latin words *geo* (meaning “earth”) and *metron* (meaning “measure”), is an evidence of this fact.

[4]See [29] for a concise assessment of ancient mathematics based on the existing evidence.

[5]We also know of some other interesting mechanical devices, although not computational, but were quite effectively used as the Chinese *south-pointing chariot* used in the Western Zhou Dynasty *circa* 1000 BCE (see [26] for a Figure(s)).

[6]The origins of decimal number system is not clear at all, e.g. see the controversial existing literature on *Bakhshali manuscript*, found in 1881.

A characteristic of *geometry* during the ancient times was its use as a not mechanical [7] but a theoretical [8] *computational machine*.

Even at times, geometric methods and algorithms were used to compensate some weaknesses of the number systems, examples of which are construction of the irrational number $\sqrt{2}$ or approximations of the transcendental number π.

Without any hesitation, Euclid's *Elements* was a masterpiece and a breakthrough written *circa* 300 BCE. The book monumentally contributed to the field, containing almost all ideas and methods used by Euclid's ancestors, and it was so well-written and complete that was copied a number of times, a reason that it reached our hands almost intact. The *Elements* not only was well-written but also had many other original aspects, each can be considered as the starting point of a new field of study. The book contained *algorithms*, as the famous Euclid's algorithm to find the greatest common divisor of two natural numbers, still in use. It also presented the first *axiomatic* approach to geometry and an abstract way of logical thinking, way ahead of its time. We do not know much about Euclid's life, but we know for sure that he used to teach at the great *Library of Alexandria* in Egypt, one of the largest and most significant libraries of the ancient world, with access to almost all scientific resources at the time. In this sense, and as far as the topics discussed in this book are concerned, Euclid is one of our heroes who has been the right man at the right place and at the right time.

By *circa* 500 CE, *number systems* as sophisticated *mathematical languages*, and *geometry* as the field that used to provide intuitive computational *methods* and *algorithms*, developed to the level that scientists could *calculate* and also *communicate* mathematical problems using positional number systems that worked almost efficiently using the concept of *zero* as *null* as well as a *place holder*.

1.1.2 Medieval period

There is no doubt that one of the greatest leaps in mathematical sciences at the beginning of the medieval period[9] was achieved by *Muhammad ibn Mūsā al-Khwārizmī* (see [24] for a Figure(s)). Again, not much is known about Khwārizmī's life, however, we know that he was a polymath living during the period *circa* 780 CE to *circa* 850 CE (within Islamic Golden Ages) under Abbasid Caliphate[10].

Most of Khwārizmī's scientific achievements were accomplished when he was a member of the House of Wisdom (also known as the Grand Library of Baghdad, an academy founded by Caliph al-Mámūn) in Baghdad, as the most important scientific and intellectual center at the time located at the capital of the dynasty. Hence, concerning the topic of this book, Khwārizmī is our second hero who has been the right man at the right place and at the right time, with an ease of access to almost all important scientific literature available at the time.

Definitely, Khwārizmī's first masterpiece is his treatise *al-Kitab al-Mukhtasar fi Hisab al-Jabr wa'l-Muqabala*[11] which is mainly about solving linear and quadratic equations and their applications. The word "algebra", that has become iconic worldwide in mathematics, comes from the word "*al-Jabr*" in the title, being one of the most important operations in the book, and meaning "compensation"[12] in Arabic.

---

[7] Sophisticated mechanical devices as computational machines appeared later (e.g. advanced astrolabe of ibn al-Shātir *circa* 1350 CE).

[8] It seems that ancient Babylonians, Egyptians, Greeks and Iranians (i.e. Persians) were experts to use simple mathematics (mainly using primitive number systems and simple geometry) to build monumental structures as great architects. Examples of this are the Great Pyramid of Giza (*circa* 2560 BCE), Parthenon in Greece (438 BCE), the Apadana palace at Persepolis (*circa* 500 BCE) (e.g. Ziggurats of Iraq and Iran (see [27] for a Figure(s)).

[9] We know that astronomical tables called *zīj* were available in Iran *circa* 600 BCE, e.g. *Zīk-i Shatro-ayār* [40].

[10] There are doubts about Khwārizmī's birth place, however, based on what we know either he himself or his ancestors were from Khwarizm (Khiva) in the Khorasan province of the Abbasid empire (present-day Xorazm Province of Uzbekistan).

[11] Khwārizmī wrote all his books and notes in Arabic, the dominant language of Caliphate Islamic dynasty at the time.

[12] In some references the meaning of *al-Jabr* is reported to be "uniting broken parts", where the word has the other meaning of "compensation" which is its correct usage in the Arabic mathematical literature, indicating the method of adding a positive number to both sides of an equation to cancel the corresponding negative counterpart, actually moving the number from one side to the other side of the equation.

An important feature of the middle-ages mathematics was the fact that mathematicians had started to use the concept of *variables* as unknowns, and they had begun to manipulate such variables to solve equations, although it was by using words[13] and not *symbolic algebra* at all. This approach was quite revolutionary compared to the geometric computations existing prior to it and Khwārizmī's masterpiece on algebra was the apex of this trend, showing how one may use these operations on variables to solve as well as classify linear and quadratic equations, justifying the label "father of algebra" attributed to him by some mathematicians and experts in history of mathematics.

Also, Khwārizmī's treatise *Kitāb al-Jam wa-l-Tafrīq bi-Hisāb al-Hind* [14] was a breakthrough and the major source of introducing the Indian (i.e. Hindi) decimal number system[15] using the concept of "zero", to the western world much later in 12th century when (most likely) Adelard of Bath[16] translated the text into Latin. It is interesting that Latin manuscripts are untitled, but are commonly referred to by the first two words with which they start, where in this case it read *Dixit algorizmi* ("so said al-Khwārizmī"), or *Algoritmi de numero Indorum* ("al-Khwārizmī on the Hindu Art of Reckoning"), a name given to the text by Baldassarre Boncompagni in 1857, pinpointing the term "algorithm" derived from Khwārizmī's name (for more on this great mathematician's scientific contributions e.g. see [2, 28, 55, 57, 65]).

After Khwārizmī, number systems, algebra and geometry were all developed to a new level through contributions of a number of great mathematicians in Middle East[17] by *circa* 1400 CE, mainly motivated by applications in architecture, finance or astronomy. With this blossoming of mathematical sciences, number systems were carefully analyzed and studied while, at last, the decimal number system and decimal fractions outruled its rivals and remained to become dominant to our present time. Motivated by applications, the major concern being the arithmetic algorithms needed for specific computations, mathematicians developed basic numerical/arithmetic algorithms for taking roots as well as approximating ratios. This, being a drastic change of procedure compared to the older methods developed by Greeks and Indians (i.e. Hindus) for computing square and cubic roots which were mainly geometric, gradually was led to the abstraction of mathematical computations through symbolic presentation of unknowns and polynomials as well as applying algebraic operations in a symbolic way[18] by the beginning of 14th century CE.

To mention a couple of landmarks, one may mention the contributions of *Abū al-Wāfā al-Būzajānī* (940 CE – *circa* 998 CE), *Abū Bakr ibn Muhammad ibn al-Husayn al-Karajī* (*circa* 980 CE – 1030 CE) and *Abū al-Fath Umar ibn Ibrahīm al-Khaiyāmī al-Nīshaburi* (1048 CE – 1131 CE), who in a sequence of contributions, gave rise to the discovery of the binomial theorem by Karajī, and eventually, the complete solution to the problem of extracting roots of any desired degree by Khayyam[19].

These investigations definitely improved the so called "algebraic methods", culminating in the masterpiece *Risālah fi'l-Barāhīn alā Masā'il al-Jabr wa'l-Muqābalah* of Khayyam containing the first complete solution of the cubic equation[20]. To highlight Iranian mathematicians' mastery on approximations, in particular using a sophisticated sexagesimal number system they had already developed, one may refer to *Ghiyāth al-Dīn Jamshīd Mas'ūd al-Kāshī* (*circa* 1380 CE – 1429 CE) who determined the value of 2π to 9 sexagesimal digits (equivalent to 16 decimal

[13] i.e. Words as "thing" (*shay'* in Arabic).

[14] There are doubts about the original title of this treatise since the original Arabic text is not available. See [28] for the latest version found in 1995.

[15] Brahmagupta's texts were translated into Arabic by *Muhammad ibn Ibrāhīm al-Fāzārī*, an astronomer in Al-Mansur's court, in *al-Zīj 'alā Sinī al-Arab* (or Sindhind), while an immediate outcome was the spread of the decimal number system used in the texts. Our current number system is based on the Hindu-Arabic number system and first appeared in *Brahmasphutasiddhanta*.

[16] We are not sure about the name of the person who has translated the text into Latin.

[17] There were also some distributed scientific activities during the middle ages in Europe, North Africa and China, however, the contributions were not comparable to that of Middle East, in particular during the Islamic Golden Ages, neither by quantity nor by quality, at all (e.g. see [13, 39, 42]).

[18] See the next section for the development of this trend into the first attempts for symbolic computation.

[19] Khayyam has said that he has explained in detail how to extract roots of arbitrary degree in his treatise *Moshkelāt al-Hessāb*, however, no copy of this book is available for the time being [7, 12].

[20] This book of Khayyam is definitely a masterpiece containing geometric methods of solving cubic equations, clearly showing the role of geometry as a computational tool in an advanced setting. Although, the classifications of Khayyam was not complete, but his approach is one of the first systematic and classified presentations in algebra, and among the best contributions to mathematics during the middle ages.

digits) in his treatise *Risālah al-Muhītiyyah* which was and remained the best approximation for more than two centuries. He also calculates the sine of the one degree arc (i.e. angle) correct to 10 sexagesimal places in his masterpiece *Risālah al-Watar wa'l-Jayb* which was quite accurate and suitable for astronomical purposes at that time.

One definitely ought to name *Abū al-Rayhān Muhammad ibn Ahmad al-Bīrūnī* (973 CE – 1052 CE), one of the greatest polymaths of all times, with gigantic contributions both in quality and quantity. Bīrūnī's encyclopaedic contributions in many different fields of study were, without any hesitation, a firm bridge for his pupils and followers to the vast scientific and cultural literature of the past. But, on the computational part, it is instructive to note Bīrūnī's masterpiece on mathematical geography *Tahdid Nihāyāt al-Amākin li-Tashīh Masāfāt al-Masākin* containing methods of computing longitudes and latitudes as spherical coordinates.

These contributions also influenced geometry to a large extent, but still in tight relation to computations, and moved the whole subject to a new level. Contributions of Khayyam, Bīrūnī and *Abū Alī al-Hasan ibn al-Haytham*[21] (*circa* 965 CE – 1040 CE), also known as the "father of optics"[22], led to an extensive study of conic sections. *ibn-al-*Haytham also worked on congruences and discovered what is now known as *Wilson's theorem* in elementary number theory.

Another great polymath, who sometimes is referred to as the "father of trigonometry"[23] is *Muhammad ibn Muhammad ibn al-Hasan al-Tūsī* (1201 CE – 1274 CE) who actually and classically invented the field of trigonometry by writing[24] *Kashf al-Qanā' an asrār al-Shakl al-Qattā'*.

During middle ages, there was a boost in numerical computations and approximations (mainly motivated by astronomy) using rational numbers, although the concepts of irrationals or transcendental numbers were quite vague. Also, it was understood that using symbols for variables to write equations rather than explaining them, makes life quite easier, at least for mathematicians applying algebraic operations, however, these were just at the level of syntax and could not be called symbolic computation at all.

Geometry, too, raised to a new level of sophistication, not as a pillar in mathematics as we know today, but still as a powerful computational tool, giving rise to trigonometry as a discipline mainly used for astronomical computations.

As far as foundations are concerned, the middle ages gave birth to a number of great philosophers, as Khayyam, *ibn-al-*Heytham and Tūsī. Khayyam and *ibn-al-*Heytham discussed and investigated Euclid's parallel postulate. Also, *ibn-al-*Heytham presents a thorough mathematical critique and refutation of Aristotle's notion of place (topos) in his *Risālah fi'l-Makan*.

But, it was only during 1500's and later, when mathematicians again considered the importance of concepts as "number systems", "geometry", "algebra", "logic" and "algorithm" from a drastically different viewpoint, while mathematics started to flourish to a new level of maturity in Europe.

1.1.3 Modern ages

During the middle ages Europe was essentially scientifically stagnated, if not in a state of decadence, until the fall of Constantinople (1453 CE) and starting the Renaissance in 15th century. The Renaissance itself highlighted humanism[25], logical thinking and philosophy, while the fall of Byzantine Empire was followed by a flow of eastern scholars into Europe, mainly through Italy. This was followed by the Industrial Revolution and two World-Wars, with their tremendous impact on our modern history and life, in which *computation* was no exception.

The period between the 12th century and the 15th century can be considered as the period of transition of science to Europe, when scientific texts were translated mainly from Arabic into Latin. Among those most influential, one may name Leonardo of Pisa (also known as

[21] Known as Al'hazen in the western world.

[22] For his masterpiece *Kitāb al-Manāzir*.

[23] We know that Hipparchus of Nicaea (died after 127 BCE) was a great astronomer of his time and contributed to mathematics by preparing, and not actually inventing, trigonometric tables of the lengths of chords in a circle of unit radius tabulated as a function of the angle subtended at the centre [37], while Tūsī's *Kitāb al-Shakl al-qattā'* is the first text solely written as a book on trigonometry.

[24] The book is sometimes referred to as *Kitāb al-Shakl al-Qattā'* which is not the correct title.

[25] The intellectual movement during the Renaissance.

Fibonacci) (*circa* 1170 CE – after 1240 CE) whose book *Liber Abaci* written on 1202 was the first text introducing the modern Hindu-Arabic number system to European scientists and merchants.

Universities of Paris and Oxford were founded at the beginning of 13th century where intellectual discussions has somehow moved from day-to-day practical motivations to the more fundamental questions on the concepts of "infinity" and "infinitesimals", mainly raised by physical intuitions.

Concerning our topic, the next influential scholar is the French mathematician, François Viéte (1540 CE – after 1603 CE), sometimes referred to as the "father of modern algebraic notation", who improved the symbolic presentations of equations to a new level, almost what we use today in an ordinary text in algebra[26]. His notational inventions, completing contributions of his predecessors, was influential in getting closer to a universal mathematical language which was definitely effective in writing clearer statements and proofs. Viéte also wrote a couple of books, among which, *In Artem Analyticem Isagoge* written in 1591 was quit similar to a modern elementary text in algebra.

The next distinguished turning point in our path is based on contributions of René Descartes (1596 CE – after 1650 CE) as a follower of Viéte, often called the "father of modern philosophy". Descartes' impact was revolutionary both mathematically and philosophically, a gist of which can be traced in his *Discourse on Method* written in 1637. His mathematical invention of *analytic geometry* in 1619 provided a deep connection between geometry and theory of equations, affecting geometry and the whole mathematics up to present time. This connection was a starting point to redefine geometry later, promoting the subject from a computational tool to a pillar of mathematical sciences, while, on the other hand, the connection supplied the theory of equations (in particular curves) with original ideas and facts known from geometry. Descartes' contributions to philosophy are manifold, but, for our purposes, it suffices to emphasize that his way of formalizing philosophical concepts as a mathematician[27], flourished later into an axiomatic approach to mathematical logic, set theory, and the whole mathematics itself, paving the road to modern theory of computation.

In relation to explicit computational techniques, John Napier's (1550 CE – 1617 CE) discovery of the concept of *logarithm* in 1614 was also a breakthrough, in particular regarding astronomical calculations, while soon calculating devices based on logarithmic concepts appeared as, the Gunter scale in 1620 or the Oughtred slide-rule in 1632 (see [25] for a Figure(s)).

Calculus is decidedly the greatest mathematical achievement of 17th century. Following a large number of contributions, eventually, Isaac Newton (1642 CE – 1727 CE) in 1687[28] motivated by problems from physics, and independently, Gottfried Wilhelm Leibniz (1646 CE – 1716 CE) in 1684 motivated by problems from logic and abstract mathematics were distinguished as the inventors of differential calculus, while both were seeking universal algorithms in Cartesian algebra. The impact of the invention of calculus was quite strong, to the extent that the whole mathematics had a shift of interest from geometry to analysis during the 18th century, in which following the contributions of some great mathematicians as Leonhard Euler (1707 CE – 1783 CE) and Joseph-Louis Lagrange (1736 CE – 1813 CE), it was Augustin-Louis Cauchy (1789 CE – 1857 CE) who based calculus on a sound definition of *limit*. The whole procedure, after two centuries, gave rise to the sound and reliable theory of *mathematical analysis* forming a perfect theoretical foundation for *numerical analysis* and theoretical assessment of computational errors.

It is also interesting to note that Leibniz was a fan of binary (i.e. digital) number system and he actually influenced the history of *calculators* by designing (in 1671) and then constructing his calculating machine called the *Step Reckoner* in 1673[29]. We know that the history of *modern*

[26] Viéte's main invention was his use of vowels for unknowns and consonants for known quantities, making it possible to see, in a much clearer way, the coefficients of an equations and their relationships to the solution. This, although, was of great help to turning the existing solution methods from a bag of tricks to a systematics and clear procedures, was also definitely not what we call *symbolic computation* today, a field which was accepted as a branch of computer science in 1960's, much later [11]. This is also a clear example of the impact of a good notation (or language) on the development of a scientific field for itself.

[27] Also, see his book *Rules for the Direction of the Mind* written in 1628. Descartes' *La Géométrie* (1637) as an appendix to *Discourse on Method* turned out to be one of the most influential books in mathematics.

[28] Newton's *Philosophiae Naturalis Principia Mathematica* (1687) is considered to be one of the most important books of modern science.

[29] However, the machine did not use binary system at all (see [53] for a Figure(s)). Samuel Morland in 1668

calculators at least goes back to Wilhelm Schickards' *calculating clock* which was designed in 1624[30]. In spite of the Thirty Year's War started in 1618 and lasted till 1648, a calculating machine called *Pascaline* (see [53] for a Figure(s)) was also designed by Blaise Pascal (1623 CE – 1662 CE) sometime about 1643.

Another interesting and related scientific event in 18th century that deserves mentioning is the discovery of a new algorithm in *calculus of variations* by Lagrange. In his letter to Euler in 1755, Lagrange describes how one may introduce a new symbol δ into the calculus and perform formally to obtain the variational equations. This, regardless of its importance within the subject, may be considered as one of the first instances of algorithmic *symbolic computation* in mathematical sciences (also see [11]).

By the end of 18th century mathematics was ripe and ready to give birth to its most important and powerful theories. On the other hand, by the beginning of the 19th century, academies as centers of excellence through Europe were gradually changing to the structure of universities, and periodicals were available to communicate scientific results easier. Most importantly, the Industrial Revolution, apart from its impacts on all aspects of the society, had already cast a shadow of automation on any concept, including *computation*. These combination of events initiated a series of scientific activities that one century later culminated into what we now call the *theory of computation*.

During the 19th century, after forming a rigorous and reliable *calculus* as a well-founded machinery, mathematics started to form its own modern disciplines as matured and sophisticated fields of study, varying from foundational subjects as *logic* and *set theory*, to *modern geometry* (algebraic, non-Euclidean), *number theory*, *differential equations* and *Fourier analysis*. This explosive and monumental improvement of mathematics itself was definitely a result of the intellectual labour work of a number of beautiful minds as, *Abel*, *Boole*, *Bolyai*, *Cantor*, *Cayley*, *Cauchy*, *Dedekind*, *Dirichlet*, *Fourier*, *Galoiś*, *Gauss*, *Hadamard*, *Hamilton*, *Jacobi*, *Kronecker*, *Lebesgue*, *Legendre*, *Lie*, *Lobachevsky*, *Riemann*, and *Weierstrass*, to name a few[31].

In relation to the theoretical foundations of computing, one may name three great mathematicians of the 19th century who provided the necessary foundational framework for what actually happened in the 1900's. George Boole (1815 CE – 1864 CE) certainly deserves the label "father of modern computer science". He elevated mathematical *symbolism* and algebra to cover *mathematical logic*, and was responsible for justifying the important role of logic in mathematics. Boole's algebraic approach to logic and his original and remarkable general symbolic method of logical inference, clearly explained in his *Laws of Thought* [3], found indispensable applications within and outside mathematics, in particular *probability*, *computer science* and *computer engineering*. The structure Boolean Algebra carrying his name, is one of the most fundamental structures in mathematics, logic and computer science[32].

Gottlob Frege (1848 CE – 1925 CE), the founder of modern mathematical logic, was responsible for a one more basic step forward toward mathematical formalism. Frege for the first time, and in a modern sense, introduced a system of mathematical logic in his *Begriffsschrift*, and also he was responsible for the invention of quantifiers and variables in modern logic. He also presented the first clear separation between the formal characterization of logical laws and their semantic justification. Frege's contributions were not well-received at his time by their contemporaries, however, his views and presentations were found to be fundamental (although contraversial too) later[33].

Giuseppe Peano (1858 CE – 1932 CE), was a key figure in the axiomatization of mathematics and was a leading pioneer in the development of mathematical logic. In a way he somehow finalized the symbolism project needed for mathematics to plume upon, and paved the way for the rest of axiomatizations in mathematics that followed. Peano's axiomatization of natural-number arithmetic is still unchanged after more than a century, in which, he funda-

designed an adding machine that did not use decimal number system.

[30]The machine was reconstructed based on a detailed description of it found in a letter of Schickard to Kepler, while we know that the prototype was destroyed in a fire (see [53] for a Figure(s)). We also know that Leonardo da Vinci (1452 CE – 1519 CE) also sketched plans for a calculator that were sufficiently complete and correct for modern engineers to build a calculator on their basis.

[31]For more on the history of mathematics see e.g. [].

[32]Boole had some other important contributions to differential equations and analysis too.

[33]Frege's contributions and views were discussed and praised later by Bertrand Russell in *The Principles of Mathematics* (1903).

mentally observed that the *infinite* set of natural numbers may be defined recursively by a *finite* set of symbols and rules, acknowledging Dedekind's priority in observing such definitions [20]. This method of providing finite descriptions of infinite sets by inductive definitions, turned out to be the core concept in theory of computation later (for more on Peano and the history of modern logic e.g. see [18, 32, 62]).

On the other hand, motivated by the Industrial Revolution, and urgent need for automatic devices that at least could do fast repetitive operations, Charles Xavier Thomas de Colmar (1785 CE – 1880 CE) built his *Arithmometer* in 1820 as the first commercial mass-produced calculating device. However, as far as the concept of a real *programmable computer*[34] is concerned, it was Charles Babbage (1791 CE – 1871 CE) who first dreamed about such a project. He did his best to raise the money needed for the project and even constructed a simplified version of his computer which he called the *Difference Machine* in 1832, but he was not fortunate enough to be able to construct a full version of his computer, the *Analytical Engine* [9] (also see [53] for a Figure(s)).

Regarding the concept of a *program*, one may mention the weaver machine of Joseph-Marie Jacquard (1752 CE – 1834 CE) called the *Jacquard loom* invented in 1804-05. Although the device was not a calculator, but the mechanism was effectively programmable using punched cards. Moreover, Ada King, countess of Lovelace (1815 CE – 1852 CE), the daughter of the poet Lord Byron and an associate of Charles Babbage, is supposed to be the first *programmer* who provided programs for Babbage's *Analytical Engine*, although, the machine did not exist physically[35].

The first half of the twentieth century is the period in which some parallel sequences of events eventually gave rise to a coalition of ideas and contributions that later formed what we call "theory of computation". A very concise and clear account of the theoretical part of the history of computability, related to this period, is provided in [60, 61] from an expert's viewpoint[36], hence, we just emphasize on some important turning points and concepts in what follows.

By the beginning of the twentieth century, David Hilbert (1862 CE – 1943 CE) [56], was the last hero dreaming about the grand project of axiomatization of mathematics, and even more, he was dreaming about consistency proofs only using *finitary* methods[37]. Hilbert's dream (later known as *Hilbert's program* [50]) gradually got matured by Hilbert's choice of geometry as one of the most developed mathematical theories of the time, while he axiomatized geometry in his *Foundations of Geometry* (1899) and used to believe that it is possible to provide a consistency proof using plane geometry. Soon it became clear that in his approach such a proof needs an axiomatization of analysis (i.e. real-number systems) that Hilbert provided in 1900, however, the consistency verification faced difficulties at the time. Hilbert's address to the International Congress of Mathematicians in 1900 is definitely one of the most important events in the history of mathematics, since the 23 problems posed by Hilbert essentially oriented mathematical research activities for a century or more, in which the second problem was dedicated to a direct consistency proof of mathematical analysis. Hilbert's 10th problem (the *Entscheidungsproblem*) on whether a given Diophantine equation has a solution, turned out to be one of the motivating problems in theoretical computer science, later solved by Yuri V. Matiyasevich in 1970 [46].

The monumental collaboration of Bertrand Russell (1872 CE – 1970 CE) and Alfred North Whitehead (1861 CE – 1947 CE) in their masterpiece *Principia Mathematica* (1910–13), although very hard to read, provided the necessary foundational and logical basis, while Hilbert also turned to foundations of logic at least during the period 1917 to 1921.

In a very simplified form, Hilbert's quest for finitary consistency proofs had very close relations to foundations of theoretical computer science and computability as we see them

[34] Before 1920's the term *computer* used to refer to people whose job was to calculate or solve various equations, as an occupation.

[35] In 1843 she provided an annotated translation of an article written by the Italian mathematician and engineer Luigi Federico Menabrea, *Notions sur la machine analytique de Charles Babbage* (1842). Also, the programming language Ada is named for her, and the second Tuesday in October has become Ada Lovelace Day, on which the contributions of women to science, technology, engineering, and mathematics are honoured (note that *Women in Mathematics Day* is set for 12th of May by IMU(CWM) as the birthday of the Iranian mathematician Maryam Mirzakhani (12 May 1977 CE – 14 July 2017 CE), the first woman who has won the Fields medal (2014).).

[36] The interested reader is strongly advised to read these academic texts.

[37] Also see [6].

today, but from a mathematical-logic point of view. In a sense, it seems that, Hilbert believed that finiteness conditions in deduction will somehow prohibit contradictions. Although, this turned out not to be true in general, but the concept of a *finitary proof* in Hilbert's view had a lot in common, at least intuitionistically, with what we call *computation* today.

Kurt Gödel (1906 CE – 1978 CE) announced what is now known as his *incompleteness theorems* in October 1930 [49]. The result, as one of the most breathtaking outcomes of mathematics, definitely had a great negative impact on Hilbert's program, while, most importantly, the concepts and methods introduced by Gödel initiated *recursion theory* and, without any hesitation, may be regarded as one of the starting point for *theory of computation* (e.g. see [50,60]).

Motivated by Hilbert's *Entscheidungsproblem*, Alan Turing's (1912 CE – 1954 CE) seminal paper *On Computable Numbers, with an Application to the Entscheidungsproblem* published[38] in Proceedings of London Mathematical Society in 1936 was also quite influential. The article proposed a definition for a *computer* that was directly based on the concept itself as a machine performing an *algorithm*, in which he characterized the fundamental characteristics of such a → Sec. ?? machine as *locality* and *finite presentability*. → Sec. ?? Another important aspect of Turing's contribution in this article was showing that there are *universal machines* of this type, capable of simulating all other such machines. → Chp. ?? The contribution was so sound and strong that the founders as Gödel and Church accepted it as one of the most important definitions for computability. At the same time Alonzo Church (1903 CE – 1995 CE) was working on the same problems and invited Turing to Princeton University to start a PhD (under his supervision) program in logic. On the other hand, Church's idea formalized as λ-calculus[39] was published about the same time in 1936.

The practical part of the story was strongly influenced by World War II and the Cold-War after that. As soon as Turing finished his PhD under Church's direction in 1938 he returned to England and immediately joined Government Code and Cypher School, where he was quite active during the war[40]. On the other hand, John von Neumann (1903 CE – 1957 CE), a post-doctoral of Hilbert and one other hero of our story, was also deeply engaged in wartime projects at IAS. World War II, although a humanitarian catastrophe, was a great push to accelerate the construction of first high-speed modern computers, one other source of motivations for theory of computation on the hardware side.

The modern history of computer design in 20th century starts by Vannevar Bush's analog calculator designed in 1930 at MIT, that was capable of solving certain classes of differential equations. Also, Howard Aiken at Harvard designed a generation of computers during 1937-1952 called Mark I to Mark IV[41]. The first special-purpose electronic computer, the ABC, was built by John Vincent Atanasoff and his student Clifford E. Berry during 1937-1942, where they constructed their prototype in 1939. Also, both Turing and von Neumann, made conceptual contributions to theory of computation by having indispensable roles in construction of first actual analog and digital computers. Turing was engaged in design of the ACE of which just a pilot model was built in 1950, but the first English computer seems to be the prototype, Colossus Mark 1, that was shown to be working in December 1943. John von Neumann was engaged in design of ENIAC in 1946, while the architecture is now known as von Neumann machine.

About the same time, Claude Shannon's (1916 CE – 2001 CE) master thesis *A Symbolic Analysis of Relay and Switching Circuits* published in 1940, used Boolean algebra to establish the theoretical underpinnings of digital circuits, and is believed to be one of the most significant master's theses of the 20th century. As far as the hardware is concerned, computer architecture quite developed during 1950's and by invention of transistors and microprocessors computer architecture started a new age of development in 1970's and after that. Different models of automata and other simple machines are byproducts of a large number of scientists' efforts to design digital circuits. Warren McCulloch and Walter Pitts were among the pioneers in computer design and also were among the first researchers to introduce a concept similar to finite automata in 1943. As a theoretical support, Kleene invented regular expressions and → Chp. ??

[38] There was a correction in 1937 too.

[39] See footnote of [60] on Turing's 1936 paper and its chronological order in comparison with Church's, Kleene's and Post's articles in 1936, as well as a very precise discussion about Turing's and Church's these.

[40] After the end of World War II Turing also had some contributions to artificial intelligence [4].

[41] Mark I was developed and constructed in 1943-44.

proved equivalence of regular expression or limited automata in 1956, and Rabin and Scott introduced nondeterministic finite automata and proved its equivalence to finite automata in 1959 (e.g. see [58] for more on this subject).

The challenges of software development were more subtle[42]. The first compilers[43] were designed during 1950's. Although natural from today's viewpoint, but it was quite astonishing at the time that one of the most important contributions in theoretical foundations of programming and compiler design was done by the theoretical linguist Avram Noam Chomsky [14, 15], → Prt. ?? while the most important hierarchy of formal languages bear his name[44]. The challenging practical issues were dealt with by a number of engineers while the theoretical part also was developed by some theoretical-minded computer scientists as Donald Knuth who invented the LR(k) parsers and showed their relationship to deterministic context-free language [43]. → Chp. ?? Software programming had to keep up with the fast grow of computer architecture design during 1970 to 1990 which led to a whale of new ideas and technologies. Maybe one of the surprises in the subject was the *scientific* resignation of David Parnas in the middle of the Cold-War in 1985 explaining why software reliability at the time was not adequate for a project as *Star Wars* [51], which led to a boost in reliable software design methods and technologies.

During 1960's all necessary prerequisites were available for forming a new discipline. A theoretical subject with very important motivations and applications in real world that was getting ready for a sprint in electronics and computer technology. In 1965 basic concepts of computational complexity were coined when Hartmantis and Stearns defined time complexity, and at the same time, Lewis, Hartmantis and Stearns defined space complexity [33, 34]. The literature was deeply influenced by *recursion theory* that was well-established at the time. It was in 1971 that Stephen Cook proved the NP-completeness of the satisfiability problem, and right after that, Richard Karp showed the existence of many other NP-complete problems using many-to-one reduction [16, 38, 44].

> It is not easy at all to fix a birthday for an evolving scientific subject, but if there is any such birthday for *Theory of Computation*, it ought to be sometime between 1930 when Gödel announced his celebrated incompleteness theorems and 1971 in which Cook proved his NP-completeness theorem[a], when the subject was matured enough in theory and applications.
>
> [a]Author's suggestion for the birthyear: 1971 ☺

The first commercially successful personal computer, the MITS Altair 8800 whose design was based on the microprocessor Intel 8080, was released in 1975, and the whole world started to have a new experience based on computers, an event that revolutionarized our society and scientific achievements, including a huge discipline now is called *Computer Science*[45].

1.1.4 The rise of "Theory of Computation" (the last 50 years)

The classical *theory of computation* (or ToC for short) flourished as an independent scientific discipline, being a consequence of a culmination of ideas from all related and different fields in

[42]Unfortunately, it seems that the history of programming is not well-documented as it ought to be.

[43]FORTRAN (acronym for FORmula TRANslation) was developed by a team of programmers at IBM led by John Backus in 1954-57, and Grace Hopper develops COBOL in 1959-60.

[44]*Context-free grammars* are special cases of *Thue systems* (e.g. see [64]).

[45]The phrase "computer science" has gone through a very fast metamorphosis according to the rapidly accelerating developments of theory and technology of computer/algorithm design. Although, referring generally to the whole subject in the past, using the word "science" as "a particular subject that is studied using scientific methods" (e.g. see [47] and references therein), for the time being, the phrase refers to an interdisciplinary field of study in which the word "science" is used more as a complement to "engineering" referring to a discipline that is more foundational and theoretical (e.g. see [17] and references therein). At least, based on ACM categorization [1], *computer engineering* is referred to as a field of study that "typically involves software and hardware and the development of systems that involve software, hardware, and communications", while *computer science* is "currently the most popular of the computing disciplines; tends to be relatively broad and with an emphasis on the underlying science aspects", as a pair of topics in a list also containing the topics, *information systems*, *information technology*, *software engineering* and *mixed disciplinary majors*.

science and technology mentioned before. Having already fixed its theoretical and technological foundations by developments of recursion theory and algorithmic complexity on the one hand, and burgeoning of computer engineering and computer technology on the other, the theory went through an accelerating maturity phase during 1970-1990's, and has experienced an explosive growth since then.

The gigantic outgrowth of computer and information technologies, in particular, at the beginning of the 21st century, has naturally asked for a theoretical backbone to lean on, pushing the subject so far as to essentially penetrate all aspects of our social, technologic and scientific life. Without any hesitation, the current century is the realm of algorithms and their applications. Algorithmic thinking is all the rage and supersedes the older approaches in science and technology, where crucial applications naturally ask for a sound and reliable theoretical foundation.

Modern Toc is almost 50 years old and is fast growing, while its natural and inevitable applications in our real-world life has already facilitated its interconnections with a variety of other scientific and technologic disciplines, sometimes making it almost impossible to categorize or distinguish between the ToC achievements and applications themselves.

In what follows we are going to touch upon some connections of ToC to other fields as well as some iconic breakthroughs within ToC itself, to show the broad scope of impact of the theory, and also to stimulate the interested reader's sense of curiosity for possible further readings or investigations. To begin, let us mention the following quote from [67].

> The Theory of Computation is as revolutionary, fundamental and beautiful as major theories of mathematics, physics, biology, economics... that are regularly hailed as such. Its impact has been similarly staggering. The mysteries still baffling ToC are as challenging as those left open in other fields. And quite uniquely, the theory of computation is central to most other sciences. In creating the theoretical foundations of computing systems ToC has already played, and continues to play a major part in one of the greatest scientific and technological revolutions in human history. But the intrinsic study of computation transcends man-made artifacts. ToC has already established itself as an important mathematical discipline, with growing connections to nearly all mathematical areas. And its expanding connections and interactions with all sciences, naturally integrating computational modeling, algorithms and complexity into theories of nature and society, marks the beginning of another scientific revolution!

During 1970's the theory of computational complexity took its first steps after Cook's seminal article on NP-completeness, while 1980's were when the theory was preparing to take off to achieve one of its most celebrated results ever, the PCP theorem. Actually, the main concepts of *interactive proof systems*, *zero-knowledge proofs* and *probabilistic checkable proofs* gradually emerged, first independently with a cryptographic motivation by Goldwasser, Micali, and Rackoff, and also by Babai having group theoretic motivations in 1983-5, triggering a race of results whose apex was the PCP theorem proved in 1992 as one of the pinnacles of the theory of computational complexity (for more details e.g. see [16, 30, 44, 67] and references therein).

Since 1970's ToC has continued its theoretical growth based on its logical and philosophical foundations as well as its response to the needs introduced by some related disciplines. The impact between ToC and mathematics is mutual and deep. (e.g. for some hints see [10, 45, 59, 63, 67]).

An interesting highlight of the theory is the proof of *dichotomy conjecture* using algebraic methods. The fact that the proof intrinsically uses universal algebraic ideas and techniques through the concept of polymorphisms provides a very deep and interesting connection between ToC and abstract mathematics. To be a bit more precise, let us recall that the concept of a *constraint satisfaction problem* covers a vast class of natural problems arising in science and technology. A bold conjecture of Feder and Vardi [19] usually referred to as the dichotomy conjecture states that any constraint satisfaction problem is either in P or is NP-complete. After a series of results eventually a complete proof of the conjecture was stated in [8, 68].

Another aspect of the theory also related to CSP's but concerning their approximations is related to the *unique games conjecture* (UGC) [41]. The *unique game problem* introduced by Khot is sort of complete for approximation problems. In [54] it is proved that assuming

UGC, there is a canonic way of finding the best approximation algorithm for any CSP. Also the conjecture is deeply related to approximation of the *isoperimetry problem* which is deeply related to abstract geometry and data clustering (e.g. see [48]).

One other fundamental concept that deserves mentioning is that of a *oneway function*, which has emerged within ToC with cryptographic motivations and has proved to be a very deep and basic notion with strong links to pseudorandomness and hardness of computation. Existence or nonexistence of oneway functions have profound consequences both in theory and applications just like the *P versus NP* problem. Also, the concept is among the most fundamental cryptographic primitives whose existence will give rise to the existence of secure secret key cryptographic schemes (e.g. see []).

On the theoretical part, ToC is mainly influenced by a need of abstraction and modeling coming from other intersecting disciplines, usually giving rise to new fields of study as *theoretical models of computation and complexity*, *theoretical models of intelligent computation*, *theoretical models of parallel computation* and a study of *intractability and inapproximability* inspired by *mathematical optimization.*

The algorithmic revolution of 21st century has somehow forced many diverse disciplines in science and technology to mingle with ToC, having deep consequences on both sides. To name a couple of such cases, one may refer to strong connections between ToC and *statistical physics* that has given rise to breakthroughs in design and analysis of complex systems, where in particular, the impact on biological and communication systems have resulted in some technological boosts.

In foundations, close interconnections between ToC and *philosophy* actually initiated form the origins, where the connections between ToC and logic is classic and mutual, starting from classical *recursion and descriptive set theories* at the early days and flourishing to introduce some new fields of study as *automatic theorem proving and proof complexity*, *semantics of programming languages*, and *theory of concurrent programming and process algebras.* It is not far from reality if one refers to *homotopy type theory* as one of the most important theoretical subjects under study which is in the intersection of ToC, logic and abstract homotopy theory. The topic just flourished by monumental contributions of Vladimir Voevodsky in proof theory and his unfinished project for *univalent foundations of mathematics* [31, 36, 66].

Interconnections of ToC through its major topic, namely *analysis and design of algorithms*, with other topics are manifold and usually inspired by applications. In this era one may mention *theory of computational complexity*, *efficient algorithm design*, *theory of secure system design and cryptology*, *design and analysis of parallel and concurrent algorithms*, and *interactive proof systems and highlevel protocol design* as a couple of well-formed growing subjects.

Artificial Intelligence (AI) is an icon of 21st century with strong links to ToC. To name some topics initiated in ToC with this relationship, one may refer to *machine learning*, *analysis and design of intelligent or evolutionary algorithms*, *presentation and analysis of vague data*, and *efficient algorithms in data science and big data technology.*

On the other hand, ToC is developing within itself and introduces new computational concepts inspired by what it receives from other disciplines. To name a few one may mention *nonuniform models of computation*, *cellular automata*, *quantum computing*, *neural networks*, *DNA computing*, and *distributed computing and multiagent systems.*

Maybe the most important and flourishing part of ToC which has been developed through the tremendous applications of networking and network models are the subjects formed and developed within different subfields of ToC in this regard. To name some of these mutual interactions, one may definitely refer to the theory of *communication networks* in which ToC has contributed both in modeling and analysis. *Communication complexity* and *network analysis* containing different algorithms related to basic problems in *network theory* as routing, partitioning (i.e. community detection) or more sophisticated algorithms related to *ad hoc* or *sensor* networks are among the subjects that have emerged through applications.

On the other hand, *coding and information theory* as well as *cryptography and design of secure systems* have always been among the subjects that have influenced ToC through their need and also contributed to ToC itself in many different and profound ways. From the point of view of applied sciences, *data science* is among the central scientific topics of 21st century, with its very natural interconnections to ToC. It is quite interesting that *big data* and *cloud computing* technologies have introduced very interesting and challenging problems to ToC where design of very efficient algorithms running at most in $O(n \log n)$ time as well as analysis and design of distributed algorithms are among the topics which are developing in relation to such needs.

Bibliography

[1] ACM, *Curricula recommendations*, https://www.acm.org/education/curricula-recommendations. 12

[2] Abdullah al-Daffa', *The muslim contribution to mathematics*, Croom Helm, London, 1977. 6

[3] George Boole, *An investigation of the laws of thought*, Walton & Maberly, 1854. 9

[4] Jonathan P. Bowen, *Alan turing: founder of computer science*, Journal of the British Society for the History of Mathematics (2015). 11

[5] Carl B. Boyer and Uta C. Merzbach, *A history of mathematics*, Wiley, 2011.

[6] Encyclopaedia Britannica, *Nicolas bourbaki*, (2017), https://www.britannica.com/topic/Nicolas-Bourbaki. 10

[7] ______, *Omar khayyam*, (2019), https://www.britannica.com/biography/Omar-Khayyam-Persian-poet-and-astronomer. 6

[8] Andrei A. Bulatov, *A dichotomy theorem for nonuniform CSPs*, Proceedings of 58th annual IEEE Symposium on Foundations of Computer Science, 2017, pp. 319–330. 13

[9] Martin Campbell-Kelly, *Computer: A history of the information machine*, Taylor & Francis, 2018. 10

[10] James Carlson, James A. Carlson, Arthur Jaffe, and Andrew Wiles, *The millennium prize problems*, Clay Mathematics Institute and American Mathematical Society, 2006. 13

[11] Bob F. Caviness, *Computer algebra: past and future*, Journal of Symbolic Computation **2** (1986), 217–236. 8, 9

[12] Karine Chemla, *Similarities between chinese and arabic mathematical writings: (I) root extraction*, Arabic Sciences and Philosophy **4** (1994), 207–266. 6

[13] Karine C. Chemla, *East asian mathematics*, (2011), https://www.britannica.com/science/East-Asian-mathematics. 6

[14] Noam Chomsky, *Three models for the description of language*, IEEE Transactions on Information Theory **2** (1956), 113–124. 12

[15] ______, *Context free grammars and pushdown storage*, Quarterly Progress Report, MIT Research Laboratory in Electronics **65** (1962), 187–194. 12

[16] Stephen A. Cook, *The complexity of theorem-proving procedures*, Proceedings of the 3rd annual ACM symposium on Theory of Computing, ACM, 1971, pp. 151–158. 12, 13

[17] Amir Daneshgar, *Interdisciplinary studies*, (2009), http://math.sharif.ir/faculties/uploads/daneshgar/Lectures/Farhangestan-89-9-10.pdf. 12

[18] Szymon Dolecki and Gabriele H. Greco, *Incomprehensible oblivion on the Peano's legacy*, Antiquitates Mathematicae **6** (2012). 10

[19] Tomás Feder and Moshe Y. Vardi, *The computational structure of monotone monadic SNP and constraint satisfaction: A study through Datalog and group theory*, SIAM Journal on computing, **28** (1998), no. 1, 57–104. 13

[20] José Ferreirós, *Richard Dedekind (1888) and Giuseppe Peano (1889), booklets on the foundations of arithmetic*, Landmark Writings in Western Mathematics, 1640–1940, Grattan-Guinness (ed.), vol. I, Elsevier, Amsterdam, 2005, pp. 613–626. 3, 10

[21] *Figure(s), *Abaci, e.g. see*, https://en.wikipedia.org/wiki/Abacus. 4

[22] ———, *Antikythera, e.g. see*, https://en.wikipedia.org/wiki/Antikythera. 4

[23] ———, *Astrolabe, e.g. see*, https://en.wikipedia.org/wiki/Astrolabe. 4

[24] ———, *Kharazmi-image, e.g. see*, https://www.britannica.com/biography/al-Khwarizmi/images-videos. 5

[25] ———, *Slide-rule, e.g. see*, https://en.wikipedia.org/wiki/Slide_rule. 8

[26] ———, *South-pointing chariot, e.g. see*, https://en.wikipedia.org/wiki/South-pointing_chariot. 4

[27] ———, *Ziggurat, e.g. see*, https://en.wikipedia.org/wiki/Ziggurat. 5

[28] Menso Folkerts, *Die älteste lateinische schrift über das indische rechnen nach al-Khwārizmī*, Bayerische Akademie der Wissenschaften, München, 1997. 6

[29] Menso Folkerts, *et al.*, *Mathematics*, https://www.britannica.com/science/mathematics (2019). 3, 4

[30] Lance Fortnow, *The computational complexity column*, 2003, http://people.cs.uchicago.edu/~fortnow/beatcs/column80.pdf. 13

[31] D. R. Grayson, *Obituary - Vladimir Voevodsky*, Nature **551** (2017), 169, https://www.nature.com/articles/d41586-017-05477-9. 14

[32] Leila Haaparanta, *The development of modern logic*, Oxford University Press, 2009. 10

[33] Juris Hartmanis and Richard E. Stearns Philip M. Lewis, *Hierarchies of memory limited computations*, Proceedings of the Sixth Annual IEEE Symposium on Switching Circuit Theory and Logical Design (1965). 12

[34] Juris Hartmanis and Richard E. Stearns, *On the computational complexity of algorithms*, Springer (1965). 12

[35] Luke H. Hodgkin, *A history of mathematics: From mesopotamia to modernity*, OUP Oxford, 2005.

[36] Institute for Advanced Study, *Univalent foundations of mathematics - program description*, https://www.ias.edu/math/sp/univalent/goals. 14

[37] Alexander R. Jones, *Hipparchus*, (2018), https://www.britannica.com/biography/Hipparchus-Greek-astronomer. 7

[38] Richard M. Karp, *Reducibility among combinatorial problems*, Springer (1972). 12

[39] Victor Katz, *The mathematical cultures of medieval Europe*, https://hal.archives-ouvertes.fr/hal-01349229/document (2016). 6

[40] Edward S. Kennedy, *A survey of islamic astronomical tables*, Transactions of the American Philosophical Society **46, part 2** (1955), 123–177. 5

[41] Subhash Khot, *On the power of unique 2-prover 1-round games*, Proceedings of the 34th annual ACM symposium on Theory of Computing, ACM, 2002, pp. 767–775. 13

[42] Wilbur R. Knorr, *et al.*, *European mathematics during the middle ages and renaissance*, https://www.britannica.com/science/mathematics (2019). 6

[43] Donald Knuth, *On the translation of languages from left to right*, Information and Control **8** (1965), 607–639. 12

[44] Leonid A. Levin, *Universal search problems*, Problemy Peredachi Informatsii **9** (1973), no. 3, 115–116, English translation, *Problems of Information Transmission* 9(3): 265–266. 12, 13

[45] Richard J. Lipton, *The $P = NP$ question and Gödel's lost letter*, Springer US, 2014. 13

[46] Yuri V. Matiyasevich, *Diophantain sets*, Usp. Mat. Nauk **27** (1972), 185–222, Translation 124-164. 10

[47] James W. McGuffee, *Defining computer science*, ACM SIGCSE Bulletin **32** (2000), 74–76. 12

[48] Assaf Naor, *L_1 embeddings of the Heisenberg group and fast estimation of graph isoperimetry*, Proceedings of the International Congress of Mathematicians, Hyderabad, ICM, 2010, pp. 1549–1575. 14

[49] Stanford Encyclopedia of Philosophy, *Gödel's incompleteness theorems*, https://plato.stanford.edu/entries/goedel-incompleteness/. 11

[50] ———, *Hilbert's program*, https://plato.stanford.edu/entries/hilbert-program/. 10, 11

[51] David L. Parnas, *Software aspects of strategic defense systems*, Communications of the ACM **28** (1985), 1326–1335. 12

[52] Kim Plofker, *Indian mathematics*, (2016), https://www.britannica.com/science/Indian-mathematics. 4

[53] William M. Pottenger, *et al.*, *History of computing*, https://www.britannica.com/technology/computer (2019). 3, 8, 9, 10

[54] Prasad Raghavendra, *Optimal algorithms and inapproximability results for every CSP?*, Proceedings of the 40th annual ACM symposium on Theory of Computing, ACM, 2008, pp. 245–254. 13

[55] Roshdi Rashed, *The development of arabic mathematics: between arithmetic and algebra*, Springer, 2013. 6

[56] Joan Roselló, *Hilbert, Göttingen and the development of modern mathematics*, Cambridge Scholars Publishing, 2019. 10

[57] Fredrick Rosen, *The algebra of Mohammed ben Musa*, Kessinger Publishing, 2004. 6

[58] Arto Salomaa, Derick Wood, and Sheng Yu, *A half-century of automata theory: Celebration and inspiration*, World Scientific, 2001. 12

[59] Michael F. Sipser, *The history and status of the P versus NP question*, Proceedings of the 24th annual ACM symposium on Theory of Computing, ACM, 1992, pp. 603–618. 13

[60] Robert I. Soare, *Computability and recursion*, Bulletin of Symbolic Logic **2** (1996), 284–321. 10, 11

[61] ———, *Turing computability: theory and application*, Springer-Verlag, Berlin, Heidelberg, 2016. 10

[62] Nikolai I. Styazhkin, *History of mathematical logic from Leibniz to Peano*, Massachusstes Institute of Technology, 1969. 10

[63] Audrey Terras, *Zeta functions of graphs: A stroll through the garden*, Cambridge Studies in Advanced Mathematics, Cambridge University Press, 2010. 13

[64] Axel Thue, *Selected mathematical papers of Axel Thue*, Universitetsforlaget, 1977. 12

[65] Bartel L. van der Waerden, *A history of algebra: From al-Khwārizmī to Emmy Noether*, Springer-Verlag, Berlin, Heidelberg, 1985. 6

[66] V. Voevodsky, *Homepage at IAS*, https://www.math.ias.edu/vladimir/home. 14

[67] Avi Wigderson, *Mathematics and computation*, 2019, https://www.math.ias.edu/avi/book. 13

[68] Dmitriy Zhuk, *A proof of CSP dichotomy conjecture*, Proceedings of 58th annual IEEE Symposium on Foundations of Computer Science, IEEE, 2017, pp. 331–342. 13