Book Introduction by the Authors

Invited by

Kazuo Iwama

iwama@kuis.kyoto-u.ac.jp
Bulletin Editor
Kyoto University, Japan

Presentation of the book: The Power of Algorithms Inspiration and Examples in Everyday Life

Giorgio Ausiello
Rossella Petreschi Eds

"To be a good computer scientist you have to enjoy problem solving. That is what it's all about: working out the best way to do things. You also have to be able to think in a logical way: be a bit of a Vulcan. But what does that mean? It just means being able to think precisely, extracting all the knowledge possible from a situation just by pure reasoning. It's about being able to say what is definitely the case given what is already known...and it's fun to do." These words appear as the 'incipit' of a series of lectures on 'algorithmic thinking' offered on the web by the educational site 'Teaching LondonComputing' (http://teachinglondoncomputing.org/resources/developing-computation al-thinking/algorithmic-thinking).

In recent years 'algorithmic thinking' has become a paradigm of the essence of computer science. A large number of initiatives have been inspired by such paradigm. In particular a series of courses on 'algorithmic thinking' have appeared on the web and some of them are included in the high quality educational program of Coursera (e.g. www.coursetalk.com/providers/coursera /courses/algorithmic-thinking). Also some books have appeared in the market with the aim of introducing students (in some cases, high school students) and professionals with no specific background in computer science to the concept of algorithm and to the key algorithmic ideas that have contributed to the impressive development of computer systems and computer applications in recent years.

In recent years 'algorithmic thinking' has become a paradigm of the essence of computer science. A large number of initiatives have been inspired by such paradigm. In particular a series of courses on 'algorithmic thinking' have appeared on the web and some of them are included in the high quality educational program of Coursera (e.g. www.coursetalk.com/providers/coursera /courses/algorithmic-thinking). Also some books have appeared in the market with the aim of introducing students (in some cases, high school students) and professionals with no specific background in computer science to the concept of

algorithm and to the key algorithmic ideas that have contributed to the impressive development of computer systems and computer applications in recent years.

In all cases two principles are stressed by all such initiatives: (i) In our modern world, algorithms are ubiquitous and all of us, in our daily life, make use of algorithms that are embedded in a variety of systems and devices (computers, playstations, cellular phones, etc.). (ii) As a consequence of (i) all individuals, since their childhood, have to become acquainted with algorithmic thinking, i.e., with the ability to formulate and solve problems in algorithmic terms.

Algorithms have been used by mankind since millennia BCE but only in the twentieth century this concept has been analyzed and its power and its limitations have been understood. In fact, it is thanks to computers that algorithms have found a place in our everyday life and the word 'algorithm' has become familiar to a large part of the population. Already in 1977 Donald Knuth wrote:

> *Until ten years ago the word algorithm was unknown to the vast majority of educated people and, to tell the truth, there was little need for it anyway. The furiously rapid development of computer science, whose primary focus is the study of algorithms, has changed this situation: today the word algorithm is indispensable.*

Nowadays, when you open a newspaper, you will likely find the word algorithm. In economics there is "the algorithm that improves access to credit"; in medicine, the "hypertension algorithm" is used to identify the most appropriate therapeutic strategy for each patient. "An algorithm to repair the memory" would act on short-term memories and has been topic of interest in both biomedicine and military contexts. "Filibustering algorithms" are used to automatically generate millions of amendments with the goal to stop the approval of a law in the Parliament. Moreover, just "put an algorithm in the engine" to generate an automotive scandal. Even algorithms to find the "ideal partner", to "read the emotions on your face", to conquer the "ideal weight" or even algorithms for "how to live an happy life" have been proposed. Finally, there are those who call "algorithm" a music festival. This common interest in the word/discipline certainly arose from the growing awareness that it is the part of the information technology, that we do not directly handle, that pervades our lives. Thus, it should be made more sophisticated to run any computer gadgets (booking a plane ticket, making a secure transaction at the cash dispenser of a bank, searching for information on the web, zipping or unzipping files containing music or images, etc.).

At the same time, the great interest raised by algorithms and their presence in so many phases of our social and private life has produced alarm up to the point that we have seen articles in newspapers shouting: "Down with algorithms", motivated by the fear that a "dictatorship of algorithms" could be created to run all human affairs.

Against such alarmed points of view, computer scientists have simply to stress the positive side of algorithmic thinking and to make clear that the formalization of a problem and the definition of an algorithm for its solution require a mental effort that leads, generally, to a better understanding of the problem itself. For this reason, algorithms are in a sense a magnifying glass that enhances our ability to understand reality in any application domain.

Let us again cite Knuth, who in his book *'The Art of Computer Programming'*, states that:

if it is true that one does not truly understand a problem in depth until one has to teach it to someone else, it is even truer that nothing is understood more completely than something one has to teach to a machine, that is, than something which has to be expressed by way of an algorithm.

The idea to explain to a large non-specialized audience (high school students, junior students in sciences and engineering, professionals experts in other domains) what algorithms are and what is their role in our society and in our daily life has been the basic motivation that led us to publish this book with the contribution of some of the top Italian experts in algorithms. We decided to consider a few paradigmatic problems that share two characteristics: we run into them almost every day, consciously or unconsciously, and their solution exploits advanced algorithmic ideas.

The book consists of ten chapters, divided in two parts. The first part (chapters 1-3) is conceived to provide non-specialized readers with general information regarding the design of algorithms and the analysis of the complexity of problems. The second part presents six different applications (chapters 4-9) which we encounter daily in our work or leisure routines. For each of these applications the conceptual and scientific bases upon which the algorithm used is grounded are revealed and it is shown how these bases are decisive as regards the validity of the applications dealt with. The book concludes with a different format, that of the dialogue. Chapter 10 illustrates how randomness can be exploited in order to solve complex problems and its dialogue format has been deliberately chosen to bring out how discussion of such issues are part of the daily life of those who are working in this field.

The text, although a logical continuity brings the reader from cover to cover, was written so that each chapter is self-consistent and can be read independently. Such choice has been made for two reasons. On one side, we wanted to allow University or High School teachers to concentrate on some specific problem without requiring to the students a complete reading of the volume. On the other side, we wanted to attract the interest of colleagues or professionals who are familiar with technologies and applications but who wish to deepen their knowledge of principles and techniques underlying the design of efficient algorithms.

Let us now give an overview of the content of the various chapters.

In the first chapter, *Algorithms. An Historical Perspective by Giorgio Ausiello,* it has shown examples of algorithms used in various historical periods, starting with early Babylonian and Egyptian algorithms and proceeding with the well known Greek example of Euclid's algorithm for the computation of the least common divisor of two integer numbers. Then an important space is given to Al-Khwarizmi. His role in the advancement of computing has been so relevant that his name has been used to denote in general any computational procedure. An important role in the history of algorithms has been also played by the people who disseminated the work of Al-Khwarizmi in Europe and in particular, among them, by Leonardo Fibonacci to whom we owe, beside the well-known series of numbers deriving from the solution of one of his mathematical puzzles, important innovations in commercial computing and accounting. Finally, after other examples of algorithms of non-numerical nature (such as the construction of magic squares and the visit of labyrinths) we have attempted to provide a concise illustration of the philosophic and mathematic trail that has brought, in the last century, to the discovery that there exist problems that cannot be solved by any algorithmic procedure, allowing to establish in such way, at the same time, the power and the limitations of algorithmic computing.

In Chapter 2, *How to Design an Algorithm by Rossella Petreschi,* the line to be followed in order to design a "good" efficient algorithm is illustrated. Integral to this, why it is essential to find the best way to abstract, represent and organize the information available about the specific problem to be tackled is explained. So, an introduction to the mathematical concept of a graph is provided since for several subsequent problems in this book the search for an efficient solution entails the modeling of information in the form of graph. In particular, the two classical problems of the "Eulerian circuit" and of the "Hamiltonian cycle" are presented and it is highlighted that, although they are very similar as regards to their formulation, they are very different as regards to their degree of difficulty. Further two examples of algorithmic techniques are introduced: the "backtrack technique" by way of the puzzle of finding how to place eight queens on a chessboard, and the "greedy technique" by way of the minimum spanning tree problem. This chapter is concluded providing a step-by-step summary of the whole process for designing a good algorithm. The binary search procedure is used by way of illustration because, although simple, allows to show all the characteristics to which an algorithm must respond (finiteness, effectiveness, definiteness) and all the stages of the design (validation, proof of correctness and complexity computation).

Chapter 3, *The One Million Dollars Problem, by Alessandro Panconesi*, is devoted to the role of complexity in problem solving and algorithm design. In particular, the chapter develops some complexity concepts already introduced in Chapter 2 and examines more in depth the properties of the two classes P and NP. Then the well known P vs NP question is presented as one of the seven problems for the solution of which, in the year 2000, the Clay Mathematical Institute has announced a one million dollars prize. The relevance of the question P vs NP and of the consequences of a positive or a negative answer to such question are then presented in terms that can be appreciated by non specialists.

Chapter 4, *The Quest for the Shortest Route by Camil Demetrescu and Giuseppe F. Italiano* digs into one of the most fundamental problems of all times: "Finding the shortest route". Whenever we look for driving directions, surf the Web, send emails, or interact with our contacts in a social network, we are, perhaps unwittingly, exploiting the efficiency of an underlying shortest path algorithm. Since the earliest civilizations, artists, scientists, and philosophers have investigated the role of the least cost principle, observing that Nature itself often tries to choose the shortest, simplest, or quickest way to achieve its goals. This chapter reviews the basic ideas at the heart of shortest path algorithms, tracing their evolution to the roots of human thinking. The pioneering work by early computer scientists who first used computers to perform useful tasks represents a milestone in our fascinating journey. Legendary scientist Edsger W. Dijkstra, since his early times as a programmer at the The Mathematisch Centrum in Amsterdam, paved the road to a series of inventions that are at the heart of our modern approach to the quest for the shortest route. The chapter moves from the fundamental algorithmic intuitions in the 1940s to the later refinements that contributed to the disruptive technological advances of the last decades in fields such as transportation, planning, and telecommunications.

Among the algorithms that hundreds of millions of people in the whole world use several times per day are the algorithms that allow fast access to the Internet and fast search on the web. To such algorithms is devoted Chapter 5, whose title is *Web Search* and that has been written by *Paolo Ferragina and Rossano Venturini.* Despite the fact that every day so many users execute them, these algorithms are known only to a tiny minority of specialists and therefore the approach presented in this chapter is particularly interesting for explaining how they reach the efficiency that is needed to satisfy the users. After introducing the two graphs that provide the mathematical models of the Internet and of the web, the authors analyze the various software components that allow information searching in the web: the browser, that reaches the web pages requested by the user and maintains them on proximity nodes of the internet for efficient future access, by means of clever caching strategies, and the search engine that provides all functionalities needed to retrieve the interesting pages in reply to a query and rank them according to their relevance.

Cryptography has a long and fascinating history that was born immediately after the discovery of writing. The 6th Chapter, *Algorithms for Secure Communication by Alberto Marchetti Spaccamela,* tells this history from the obscuring texts in the Egyptian hieroglyphics to modern public-key cryptography. It is underlined how, before the widespread distribution of modern computers, cryptography was practiced by few people: either soldiers or diplomats (for war reasons) or lovers (as the Kama Sutra reports). Cryptography algorithms designed in the past were ingenious transformations, but were lacking a sound mathematical basis. On the contrary, the modern approach to cryptography is based on sophisticated mathematical techniques and methodologies. In fact, the development of powerful computers has made possible to implement sophisticated algorithms that require complex calculations to encode and decode, in particular those based on functions that are easy to compute but difficult to invert. Finally, the chapter points out that the development of computers and of Internet has opened new applications of cryptography in business and society. The new sound mathematical techniques, realized for obtaining these applications, have opened impressive possibilities that were considered unrealistic before.

In chapter 7, *Algorithmics for the Life Sciences by Raffaele Giancarlo* an effort is made to explain the correlations between Information Science and the fundamental laws of Biology. Indeed, a relevant part of the incredible advancement in knowledge obtained by molecular biology, since the discovery of "the double helix", has been possible thanks to synergies with computer science and mathematics. In this chapter, some of the most relevant aspects of this cooperation are reviewed and some algorithms for bioinformatics that are recognized as reference points in a particular domain are presented. Finally, it is emphasized that, even if specialized databases and specific tools are required for managing the huge quantity of biological data, computer science and mathematics are fundamental not only for the data warehousing aspects, but even more for the analysis of those data in order to reach conclusions of relevance for biological research.

Chapter 8 is devoted to algorithmic aspects that are crucial in synchronizing electromagnetic signals in TV broadcasting. The chapter has been written by a team of experts in operations research and in its applications to telecommunications: *Fabrizio Rossi, Antonio Sassano and Stefano Smriglio.* Its title is: *The Shortest Walk to Watch TV* because at the heart of the presented techniques is an algorithm for computing the shortest path on a graph where also negative distances are allowed. In large telecommunication networks, the choice of transmission frequencies is dictated by two factors: on one side, the need to use different frequencies in order to avoid interferences; on the other side, the fact that the frequency spectrum is a limited and expensive resource. As a consequence, it is important to find strategies that allow to reduce the use of frequencies while maximizing the coverage. One of the techniques that are adopted consists in creating artificial transmission delays in such a way to compensate the different distances traveled by the signal. In a complex network this generates an optimization problem one of whose essential steps is, as we said before, the solution of a shortest path problem with negative weights.

It is probably well known that the revenues of a company such as Google essentially derive from the presentation of commercial advertisements on the screen together with the answer to user queries. In order to produce such results, Google runs on line auctions millions of times per day. This is only one of the best known applications of concepts arising by the encounter between economics and algorithms. Chapter 9, *Algorithms for Auctions and Games,* written by *Vincenzo Bonifaci and Stefano Leonardi,* addresses exactly this issue starting from the observation that, as the authors say, the exchange of ideas between the theory of algorithms and the economic theory of games is one of the most fascinating facets of computer science in the last decade. In the chapter, after providing an introduction to game theory and to the concept of Nash equilibrium, how game theory can be applied in computer science is explained and in particular the application of game theory in the choice of routing strategies in networks is addressed. In second place the use of auctions in computer science is taken into consideration, a familiar topic for millions of customers used to buy their products on eBay. More generally, the topic of mechanism design is introduced, with a particular emphasis on the design of truthful mechanisms such as the Vickrey auction and other models, and its computational aspects are discussed.

Chapter 10, *Randomness and Complexity by Riccardo Silvestri*, presents a dialogue among three people who are enthusiastic about the unexpected power of random choices. From a database search to traffic analysis on the Web, from data mining to cryptography, several hard problems can be efficiently solved with the help of probabilistic algorithms, but random choices are also very elusive. If they were too powerful, some cryptographic algorithms used daily would no longer be trusted. At the heart of this phenomenon, the interplay between causality and complexity creates a fascinating world that is still largely unexplored. The conversation begins discussing a seemingly trivial problem which leads to speak of random search, a basic technique of probabilistic algorithms. Then, the discussion moves on the more sophisticated technique of probabilistic counting. Exploring it, general questions about the ways of using probabilistic algorithms are introduced, starting from the Miller-Rabin primality test. To shed light on how a computer might do random choices, the debate hinges on Kolmogorov complexity, pseudorandom generators and the power of random choices discovering that, if they were too powerful then many cryptographic protocols, like RSA, would not guarantee an adequate security.

After having provided this brief illustration of the content of the book, as a conclusion we would like to reaffirm that to read this book no deep background in computer science is required: it is simply sufficient to have some familiarity with basic mathematical concepts and notation. What is indeed required is curiosity for some important problems that have an impact on our everyday life and an attitude toward algorithmic thinking that can help to understand various approaches to their solution.