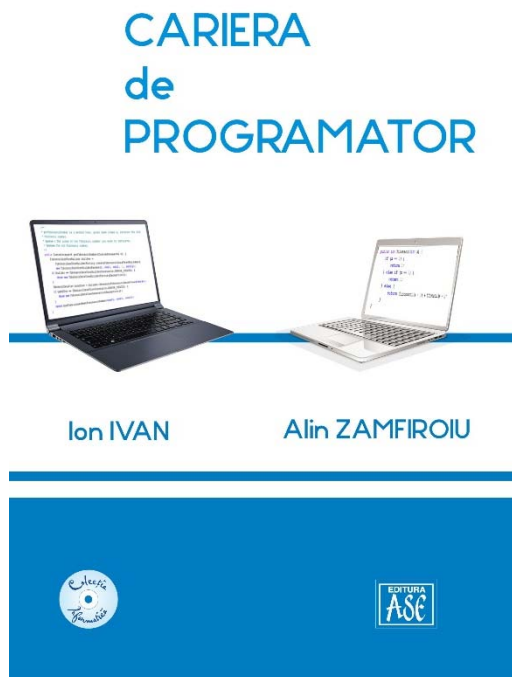


## Book review:

### *The programmer career*

by Ion IVAN and Alin ZAMFIROIU



The book „The programmer career”, written by Ion IVAN and Alin ZAMFIROIU, members of the Department of Economic Informatics and Cybernetics, was published in 2017 at ASE Publishing House under excellent graphic conditions. During the 236 pages, the authors present the most important aspects of the programmer's career, starting from the premise that his work is mainly creative work, based on a sustained documentary effort, since the field of informatics is characterized by a diminution accelerated, and the moral wear of technologies, tools, languages and equipment is dramatic.

In the first chapter entitled **Clarifications**, some necessary details are given about concepts such as programs, scheduling, programmers, algorithms, programming languages, specifications, testing, interfaces, customers, quality and cost, to define the complexity of the context in which programmers operate, knowing that all efforts related to the development of a software product are concretized only through their work.

The second chapter, **Developer's Qualities**, highlights that not every IT&C person has resources to develop a career as a programmer. An authentic programmer has to show patience,

clairvoyance, perseverance, honesty, tenacity, communicativeness, objectivity, self-evaluation, but also a permanent desire for self-indulgence through continuous training with the newest and most valuable results obtained in his field of activity.

**The road to perfection** is the chapter in which Ion IVAN and Alin ZAMFIROIU are leaning on the efforts that those who want to embrace the programmer's career must do to succeed in establishing themselves as authentic specialists whose performance is recognized by other programmers, especially as software executives implemented and executed. The road to perfection begins with shaping the personality by accepting work as the only criterion of evolution in the craft, the deepening of the specialization in order to have the necessary capacity to know all the details regarding the resources offered by the chosen subdomain. In programming work, staying in the area of generality is the main brake in perfection as a programmer. The developer must accept reality both by the speed at which technology changes, but especially by the need to move himself to the use of those revolutionary technologies. Emphasis is placed on the need for continuous learning, teamwork and the courage to accept continuous testing of the knowledge gained by participating in the acquisition of qualification certificates recognized as value in the world of specialists.

In the **IT Project Design** chapter, specific programming issues are listed in the development of software products as basic elements in funded IT projects, characterized by precise objectives, targeted target groups, limited implementation times and imposed quality parameters. The authors of the paper insist on concepts such as successful IT projects, the quality of software products, metrics and the need to develop embedded software products from the start. By its nature, the programmer's work needs to be geared toward producing high-quality products, because the software footprint is extremely dynamic and a product with a somewhat older technology, entering the market is morally outdated, products developed with much newer technologies, more productive than the product.

**The programmer's risks** are countless as routine, rapid success, persistence in an inadequate programming style, self-satisfaction, non-involvement in increasingly complex projects, empirical approach to problems, waste of resources from negligence, working in a dispersed, the perpetuation of new beginnings and the unjustified change of teams, play a negative role in the career of any programmer. These risks are the subject of an analysis based on the experiences experienced by many programmers with whom the authors have collaborated over the years.

The authors of the book have paid special attention to increasing the productivity of the work of programmers in the chapter entitled **The need to reuse**, placing a special emphasis on the concrete conditions that make operational the integration of software components already in use in new computer applications, even if they are developed with new technologies. Software re-use is itself a field of research and object-oriented technologies are meant to increase the reuse of components. There are also some limitations of reuse, especially when it comes to components whose update dynamics is very high. In the case of re-use processes, the authors identify optimization elements that highlight both the volume of components to be integrated without affecting the performance of the new product and the time when the reuse process ends.

The programmer's career spans a period that varies from a few years to a few decades, depending on the ability of everybody to resist writing programs. That is why the authors grant a wide-ranging space to the book on **Promoting Programmers**. Each software developer has its own criteria and steps to advance, but the work of programmers being precisely quantified and qualitatively quantified needs to find a real correspondent in salary but also in the position of the team in which the programmer works. There are many situations when some programmers are migrating to other companies, since their appreciation in the company is not desirable, they are not used to the maximum capacity of their knowledge, even if they are recognized in the world of specialists but also in the software market, representing a brand just.

**Programmers' typologies** are the chapter where the authors of the book detail the features of the art developers, develop database applications, write operating system components, develop web or mobile applications, but also deal with programmers who maintain, innovate and separate from those who go through all stages of the software development cycle, being like orchestras. Things have gone a long way in the job of a programmer that it is almost impossible for a programmer of a certain typology and with a particular level of performance to instantly switch to the development of another software typology with the same success.

Finally, the paper has a few **Conclusions** pages in which the optimistic tone shows that the work of the programmer remains essential in the future to develop software, even if the tools that will be further developed will take over many of its routine elements. It is known that mankind will never develop robots to build other robots of greater complexity than their own. Similarly, programming automation tools will not be able to build components of greater complexity than complexity in them, and the creativity of tools is and will always be inferior to the creativity of human programmers.

The **Programmer's Career** book reflects, on the one hand, the experience of a programming activity that extends over four decades of one of the authors from machine code programming on the Romanian computer to CIFA 101 tubes, passing through FORTRAN and COBOL on IBM 360 and FELIX C 256, reaching the PASCAL and C ++ languages on today's personal computers and web applications on the Internet. The other user is breaking down the current Android mobile applications and deploying security algorithms for them, but also doing scientific research in the field of computer science. The book is intended for both IT students and those wishing to work in the field of computer science as programmers and to look at the programmer's job in terms of its nobility, importance and value in completing the stages of the software development cycle.

**Assoc. Prof. Cătălin BOJA, PhD**

Director of the Department of Economic Informatics and Cybernetics