

Krzysztof JELLONEK  
Małgorzata KOTULSKA \*

## **NEW CONCEPTS AND TECHNIQUES IN EDUCATION OF BIOMEDICAL ENGINEERS IN COMPUTER SCIENCE**

Modern university education is very strongly based on computer techniques, which provide necessary computational power to develop specialized software. Very intuitive graphical programs became indispensable tools for word and graphics processing, data storage, and advanced computational tasks. An increasing role of information availability from the data stored on web pages and network accessible data bases can be observed. This immense potential source of information requires appropriate searching tools and algorithms. In this situation a well-scheduled track through computer science courses, for students of various specializations, is an essential issue. This problem is especially acute in education in Biomedical Engineering since application of computer methods to medicine, computational biology, and bioinformatics requires an approach that differs from classical methods. We discuss concepts and techniques in teaching computer science courses, comparing three specializations from the Faculty of Basic Problems of Technology in Wrocław University of Technology: Biomedical Engineering, Computational Physics in Science and Technology, and Mathematical Aspects of Computer Science.

*Keywords: education, biomedical engineering, computer science.*

### **1. INTRODUCTION**

It is undeniable that educating of an engineer in any specialization requires presently very strong background in computer science. However, this objective can be achieved by various methods [3, 4, 5].

Educational path through a wide variety of available computer courses with regard to their selection, content and training methods is still a debatable issue for many university committees deciding on the curriculum.

---

\* Wrocław University of Technology, Division of Measuring and Medical Electronic Instruments, 50-350 Wrocław, Poland, e-mail: kotulska@pwr.wroc.pl

Computer science contains two complementary domains (Fig.1): hardware and software. Software includes tools, techniques and theory. Software tools can be defined as specialized programs dedicated to solve or support a specific engineering task. Software tools include programs for computer-aided design, simulating physical processes, data analysis, data transmission, and all sorts of multimedia tools - for example supporting on-line presentations and lectures. A separate class of software tools of major importance is programming languages - tools for developing more specialized, usually problem-dedicated software.

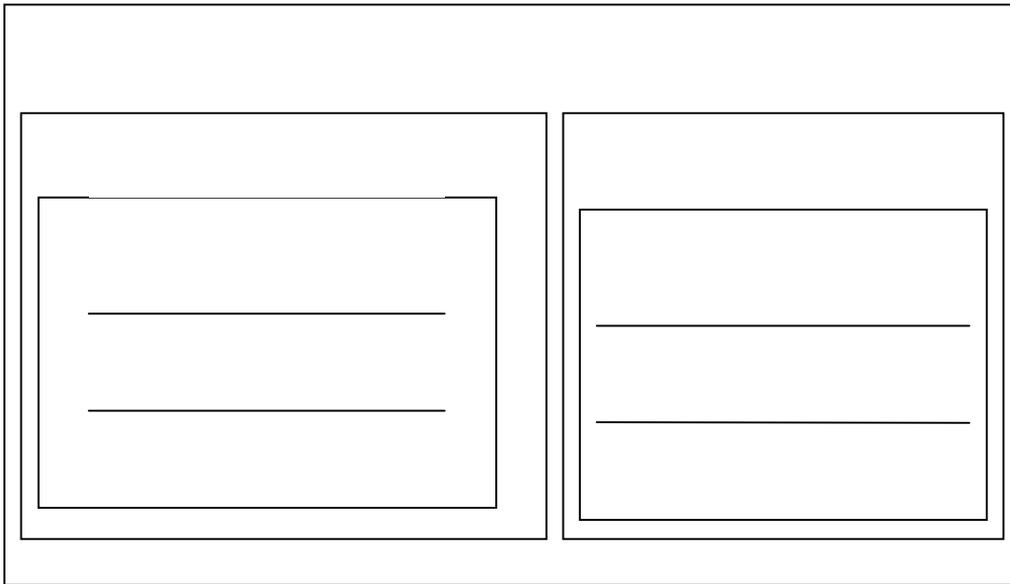


Fig. 1. Major components of computer science that should be considered in curriculum

Computer techniques include algorithmic and non-algorithmic methods necessary for optimal approach to engineering tasks. They include optimization, identification, decision support and expert systems, data base techniques, fuzzy logic, neural networks or cellular automata. All algorithmic computer techniques require theoretical basis, which provides mathematical foundation for new algorithms used mainly for data processing, storage, and especially recently data mining [9]. The theory improves existing methods increasing their speed and efficiency. It also provides completely new solutions, whose practical implementation eliminates numerous limits of the computer applications.

Hardware domain consists of three layers: computer system blocks (logic systems, microprocessors, memory, etc.), computer devices (computers, cards, printers, hubs, etc.), and computer networks. A non-electronic engineer education can be limited to

first two layers, it requires however different teaching methods. Generally, computer science can be taught in four stages:

- General outline of the problem,
- Full theoretical basis related to the topic,
- Practical knowledge necessary to informed use of the specific method,
- Practical skill for software tools.

The knowledge can be conveyed by means of several various techniques:

- Encyclopaedic lecture,
- Dedicated lecture (concerning one particular problem),
- Laboratory class,
  - individual tasks corresponding to specific problems,
  - group collaboration reflecting real-life approach for problem solving,
- Project,
- Seminar,
- Independent study based on lectures/data available in literature and internet,
- Practice in a computer company.

Designing appropriate curriculum for computer science education for future engineers, not specialised in electronics or computer science, requires addressing the following issues:

- Knowledge extent and level,
- Teaching methods and time extent,
- Optimal sequence of the computer courses.

Decision should be based on the following factors:

- Analysis of the potential employer's demands,
- Evaluation of former graduates careers,
- Prediction of the prospective progress directions in computer science technology and demands from the particular industry branch,
- Individual experience of academic researchers from the specific engineering field,
- Curriculum experience of other Universities concerning students of the same specialization.

An accurate prediction of the possible time period when certain type of knowledge will be still useful and relevant is also an important factor. It concerns mostly software tools, which are available in great variety. A student should be trained for flexibility in tools rather than very high skill for one particular type of tools. There is a good chance that they will be using different software tools in their professional career. One reason is fast aging process of the software. However, the change might be not necessary for the better. The company employing the graduate may be still attached to old tools because the change for even better and more efficient software would result in discontinuity of their operation, necessity to adapt or even lose their old resources and home made software.

## 2. THREE FACES OF COMPUTER SCIENCE EDUCATION AT THE FACULTY OF BASIC PROBLEMS OF TECHNOLOGY

All these issues were taken into consideration at the Faculty of Basic Problems of Technology in Wroclaw University of Technology, where the problem of the optimal curriculum for computer science courses is crucial especially for three specializations (Fig.2): Biomedical Engineering, Computer Physics in Science and Technology, and Mathematical Aspects of Computer Science.

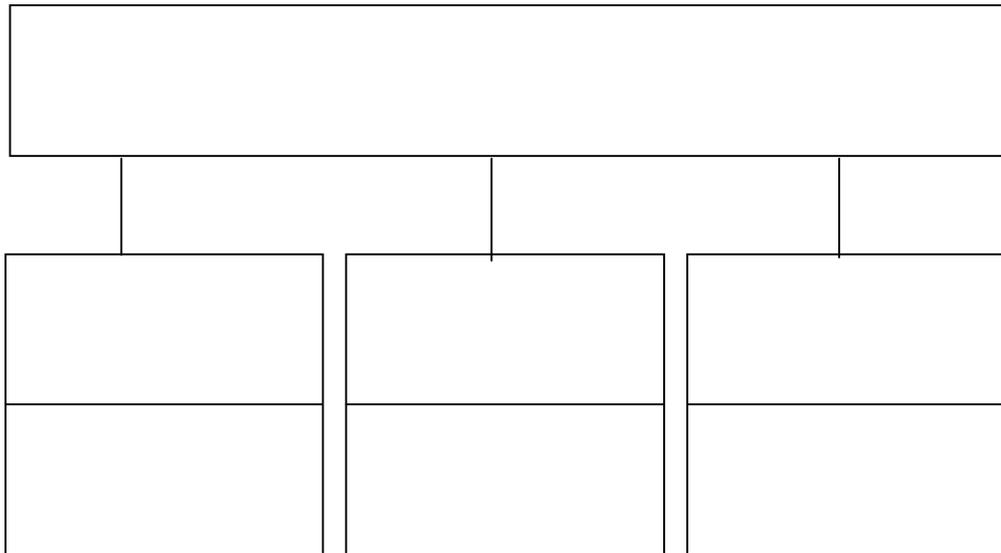


Fig.2. Three different layers of in computer science education differently stressed in three specializations at the Faculty of Basic Problems of Technology

Computer Science courses are organized into three blocks: fundamental compulsory courses, compulsory or optional specialized courses, and problem-dedicated optional courses. A more detailed comparison of the curricula for those three specializations shows differences in the approach to teaching computer science, arising mostly from different needs.

The most technological specialization is Biomedical Engineering – Medical Instrumentation, which belongs to the Technical Physics specialization group. Here, the block of basic courses includes:

- Computer Science I, II – introduction to computer science, which covers basic terms and fundamentals of programming in C language,
- Algorithms and Data Structures – a course devoted to the techniques of expressing a technical problem in terms of an algorithm. The course teaches also

optimal and correct data structuring and processing, including data structure linking, object-oriented and virtual methods.

- Application Programs – the course devoted to the most useful tool programs for word and graphics processing, spreadsheets, simple database, and Internet searching with fundamentals in web-page design. However this course is optional and can be skipped by more advanced students, for those who select it, Application Programs comes as a first computer science class.

It should be noted that a first programming language is introduced in the block of fundamental courses. An appropriate choice of the language is a debatable issue. The decision should take into account how rigorous and clear is the language structure and if it is capable of training students in a good programming style. A classical programming language Pascal was considered the best for teaching purposes for many years. However, the tendency to teach languages practically used as programming tools in industry related to students specialization proved stronger. The hope for increased employment chances was the main reason. Presently, there are three most popular languages selected by students: C++, Delphi, and Java. However Java teaches students programming in the object-oriented style right from the start of learning computer science and enables for a full internet use, it is still taken as a second language.

The family of Unified Modeling Languages (UML) [7,8,10] is also rising a great interest in teaching area. These languages are powerful in the situations in which structural model becomes insufficient to handle the complexity of huge computer systems. Along with object-oriented programming it proved successful in business and technology and seems appropriate also for teaching purposes.

It is interesting that all seriously considered languages are of algorithmic structure. Logic programming languages such as Prolog, Logo, and LISP did not prove very successful, mostly because they do not teach how to optimize a programming code in terms of time or memory use. However, these languages also have their proponents who claim that such programming style, which is the closest to human heuristic reasoning, introduces a newcomer to computer science in the most natural, intuitive way.

The block of specialized courses puts special impact on the education in Computer Measuring and Diagnostic Systems. Students take compulsory courses in Applications of Microprocessors, where smart cards and transducers are one of the main topics, Measuring and Diagnostic Systems (lecture and project laboratory). The problem-based style during laboratory classes mimics real-life commercial projects, requiring multidisciplinary approach and student collaboration within their own project group as well as inter-group communication. An exemplary project: Medical System for a General Practitioner (MSGP) gave students an opportunity for practical implementation of modern computer technology, smart transducers, computer networks, telecommunication tools such as GSM, GPRS, and other software tools [2].

The blocks of specialized and dedicated education contain also optional courses such as Computers in Medicine, Programming in Java, Object Programming, Archi-

ture of Computers, Computer Networks, Computer Graphics, Optimization Methods, Identification and Modeling of Biomedical Systems, Artificial Intelligence and Expert Systems, Neural Networks, Decision Support Systems, Computer Image Processing. A new course in Bioinformatics with special regard to algorithms specific only for biological data and data mining techniques is currently considered.

The curriculum based on these premises succeeded in this respect that most of our graduates do not have major problems with fitting into a wide variety of professions in medical equipment industry, computer industry, and telecommunication.

A different approach was applied into two other specializations: Computational Physics and Mathematical Computer Science. Both of them educate less technologically-oriented students than Biomedical Engineering. The students of those specializations feel more related to natural sciences than to technology. Here, the education should increase the ability to advance the layer of theoretical basis in computer science (students of Mathematics) or computer techniques (students of Physics) rather than train them in developing new software tools. This assumption laid the foundation for the curricula in Computer Science courses.

Physics students start their education solving simple problems in Delphi, and proceed to higher level languages used mostly for mathematical aspects of physical problems such as Mathematica, Maple and Matlab. The ability to visualize the solutions is acquired through the course in Multimedia Techniques for Physicists. Other optional courses include: Elements of Computer Simulations in Physics, Statistical Computer Packages, Numerical Linear Algebra, Object-Oriented Programming, Computer Networks, Operating Systems, Data Networks, Neural Networks, Computer Applications in Natural Sciences, Computer-Aided Design, Computer-Aided Experiment.

This type of education in computer science should provide the basis necessary to simulate physical process, analyze it, and visualize the results. Experimentalists are taught how to involve modern computer technology into their research.

Students of Mathematical Computer Science are focused mostly on the theoretical layer of the computer science. Therefore, their curriculum contains fewer courses teaching practical knowledge and engineering methods. A particular stress is put on theory. The education starts with Programming I, II courses, which are not dedicated to teach any particular programming language but fundamental properties on structural language design. More specialized courses include Object-Oriented Programming, Computational Complexity, Advanced Algorithms, Software Engineering, Data Compression, Data Mining, Operating Systems, Data Base, Network Protocols and Distributed Systems, Group Programming Project, and Computer Graphics. Optional courses include also Cryptography, Web Technologies, and problem dedicated classes where individual projects are discussed. Graduates from this specialization are supposed to support software programmers with new ideas underlying working algo-

rithms. This specialization is very new in our Faculty so its curriculum is still very changeable and the results of this type of education unknown.

### 3. CONCLUSIONS

We discussed various approaches and tendencies to education in computer science courses at technical Universities. In the era of very rapidly changing technology and industry demands outlining an optimal curriculum is a challenging task [6,4,3]. Modern tendencies were presented on the basis of three different specializations, all of them from the same, least technological Faculty of Basic Problems of Technology from Wroclaw University of Technology in Poland. These three specializations reflect three layers of Computer Science: theory, techniques, and tools. Students of Mathematical Aspects of Computer Science are trained in good theoretical basis to advance methods in Computer Science. A stress is laid on their flexibility, where particular programming tools used to test new ideas are of secondary importance.

More practical aspects can be observed in curriculum of Computational Physics in Science and Technology, whose graduates are expected to develop new techniques as well as use existing methods to improve our understanding of physical phenomena. The most technological specialization in this Faculty is Biomedical Engineering, whose graduates will be responsible for designing and implementing technological tools to support medicine and research in biology and medicine. The curriculum for computer science courses should take into account that medical and biological data are highly specific.

We conclude that generally for most engineering specializations only the layer of computer tools should be adjusted to their field, while two other layers i.e. theory and techniques can be universal. However, for specializations closer to Basic Sciences, such as those presented in the paper, a more individual approach to the curriculum is necessary in order to achieve good educational results.

### REFERENCES

- [1] ROACH S., *The impact of technology on engineering and computer science education in the 21<sup>st</sup>/sup Century: changing classroom instructional methods*, 31<sup>st</sup> Annual Frontiers in Education Conference, Impact on Engineering and Science Education. Conference Proceedings IEEE. Part vol. 1, 2001, T3C-20-2 vol. 1, Piscataway, NJ, USA.
- [2] JELLONEK K. KOTULSKA M., *Measuring and diagnostic systems in biomedical education*, Engineering Science and Education Journal 2002, 11, 1, 2 – 6.
- [3] BAUMANN M., BLUM, P., GREBE, R., SCHMID-SCHONBEIN H., *Computers in physiological practical education prerequisites and computer based training*. Computers in Education Journal, 2000, X, 3, 56–76.

- [4] ERIKSSON T., GOLLER A, MUCHIN S., *A comparison of online communication in distance education and in conventional education*, 31<sup>st</sup> Annual Frontiers in Education Conference, Impact on Engineering and Science Education. Conference Proceedings IEEE Part vol .1, 2001, T2F-20-5 vol. 1. Piscataway, NJ, USA.
- [5] CROFT A.C., DANSON M., DAWSON B. R., Ward J. P., *Experiences of using computer assisted assessment in engineering mathematics*, Computers and Education 2001, 37, 1, 53-66.
- [6] FORBES N., MESSINA P., *Computer science today in the European Union*, Computing in Science & Engineering 2002, 4, 1, 10-14.
- [7] KRISTENSEN J. T, HANSEN M. R, RISCHHEL H., *Teaching object-oriented programming on top of functional programming*, 31<sup>st</sup> Annual Frontiers in Education Conference. Impact on Engineering and Science Education. Conference Proceedings IEEE. Part vol. 1, 2001, TID-15-20 vol. 1. Piscataway, NJ, USA.
- [8] RUMBAUGH J., JACOBSON I., BOOCH G., *The unified modeling language referenc manual*, Addison Wesley 1999.
- [9] ADRIAANS P., ZANTIGE D., *Data mining*, Addison – Wesley 1998.
- [10] WEBB W., *Virtual programers build embedded code*, EDN EUROPE 2002, 33-41.