INFORMATION SYSTEMS IN MANAGEMENT

Information Systems in Management (2015) Vol. 4 (4) 285–290

ON SOME RECENT PROBLEMS IN COMPUTER EDUCATION ON ENGINEERING STUDIES

WITOLD SIKORSKI

Department of Environmental Engineering, Warsaw University of Technology

Education concerning computer tools for the future engineers is present at the universities for over 40 years and since every student of engineering faculties (not only computer engineering) obtains knowledge of computer applications and data processing tools. We are often asked why do we teach IT if the computer is commonly used and every student is well acknowledged with its applications. New problem arises as the students use computers, but do not understand its meaning in their future work. There is a gap between everyday thinking of computer application and real needs for the engineer. The aim of the presented analysis is to discuss how to fill this gap and prepare syllabus to gain students interest and present them with understanding of the tools they are about to use in their work.

Keywords: IT curriculum, computing tools for engineers

1. Introduction

Educating engineering students in the field of usage of the computer tools is present at Warsaw University of Technology for over 40 years. It is obvious that every student of all engineering faculties (not only computer engineering student) should obtain the knowledge of computer applications and data processing tools.

In the early seventieth information technology as a subject was introduced in most of the faculties of our University. It was something modern, new to the students and nobody neglected that it is needed in the future work. It was mainly binary mathematics and programming then.

Today we are often asked why we should teach IT when a computer is commonly used and every student should be acknowledged with its applications. However we come over a completely new problem – the students use computers, but do not understand its real meaning in their future work. There is a gap between everyday thinking of computer application and real need for the engineer. There is a problem of digital literacy and “telepresence” as opposed to presence in the real world [1].

Regular contact with short visual information since early childhood might in a long term influence the perception ability of peoples’ brain [2]. New “digital generation” is at our doorstep.

2. Filling the gap

The aim of the presented analysis is to discuss how to fill this gap. How to prepare syllabus to gain students interest and present them with understanding of the tools they are about to use in their work. The problem concerns several basic educational topics:

- How to develop an understanding of commonly used numerical methods for solving engineering problems and the ability to appropriately apply numerical methods with the knowledge of some of the limitations of such methods.
- How to manage large portions of data with commonly used tools as spreadsheets and databases.
- How to develop a knowledge of basic programming concepts and the ability to write simple programs to apply them as a tool in applications like CAD and other engineering tools.
- How to select the information obtained via Internet – the choice between the good and the rubbish.

The aims are quite clear, but incorporating them to the syllabus is rather difficult. The syllabus should be aimed at understanding not at practical skills in concrete software. We do not know what software will be available in 3 to 5 years from now when the students will get their degree. So we need to concentrate on broad concepts using specific tools only as an illustration and example.

An approach of the students is rather negative – they assume that they are well educated in computing techniques so they neglect the need of general education. Reality is quite different – they are illiterate even in terms of using such applications as word processor. Their thinking is limited to search engines (like Google) and learning menus of certain applications, rather than on reasoning and general need of the tools.

The problem is also in a very low level of school education. Our experience shows that large percent of the first year students is not aware of real strengths of modern computer application on one hand, over estimating Internet techniques on the other. The approach that everything is easy and “done by the program”, so there is no need to study it. It is for them like using a hammer and saw, so they do not accept the need of understanding the complexity and theory needed to use the tools properly.

3. Building a curriculum

The consequence is that we are often approached by the students with the following questions:

- Why do we need to learn IT basics?
- Why do we waste time on general information of data encoding, search and validity?
- Do we at all need to learn programming as we will not build the software?
- What is the reason to learn the general information on tools like spreadsheet or math tools (like Matlab)?
- Why we should learn CAD basics instead learning it in practice during construction or design classes?

We include these questioned topics in our curricula. An analysis of the curricula at most of the technological faculties shows similar approach. So the problem is not what to include, but how to present the contents to the students, so that during 30 to 60 hours of classes they get no only the knowledge, but also get convinced that they did not waste their time. We should analyze the attitude of the students as it is presented in [3]. This is something really important in the course of curricula improvement.

3.1. Curriculum – lectures

Generally we should include the content of IT for engineers as a general subject, similar to mathematics and physics, trying to avoid an application oriented approach. This implies the role of a good lecture at the start of the classes. The lecture should consist of three general blocs:

- Information encoding: a very important topic showing that all information available conforms to some general rules. There is an important element of breaking encoding into certain levels to show the consistency of the applied methods.
- Data obtaining, storage and processing: a general approach to data, their validity, ways of storage and processing methods. Some references to data management tools might be useful, but the emphasis should be put on the availability and validity.

- Programming basics: a program as a way to present an algorithm, not as a software development tool. Program and programming should be shown as the last step in the analysis of the problem and one of many tools to make use of an algorithm. Understanding the relation between an engineering problem and a solution is key element.

The lecture will thus help to show the background of information storage and processing, emphasizing the role of IT with some depths in the opposition to plain clicking “solve” button in any application.

The practical part of the curriculum must be oriented on general implementation of the above lecture topics. Generally we may divide the practical classes into similar groups of topics. We need some presentation of data encoding, and a large block of engineering tools enhanced by the elements of engineering software.

3.2. Curriculum – the practical part

Let’s try to look at the types of applications and their role in the curriculum. We may classify the tools into three groups: general data processing, general engineering tools and specialized tools for engineers including different types of computer graphics. We will now look at these groups from the point of view of the curriculum keeping in mind the questions and doubts that our students still have.

Practical classes should be built upon an idea of three steps to the solution – presentation of the problem, choice of a tool (appropriate software) and finding specific functions. The students cannot base on specific software including specific version, memorizing menus, steps and the names of the functions. They have to search for the tool basing of a general idea what he or she is supposed to obtain as a result.

It is preferred to split the practical course into 3 or 4 semesters. The students should be gradually introduced to the tools that are more and more oriented on solving practical engineering problems. We propose the following groups of applications:

- Spreadsheet as a tool of basic data analysis with the emphasis on filtering and aggregation of the data. It is especially useful for the specializations where the large number of data has to be considered. It may be combined with programming basics, pointing out the possibility of programming elements included in such tools like spreadsheet (i.e. Visual Basic for Applications).
- Engineering tools to perform calculations, solve equations and various numerical problems. (i.e. tools like Matlab or MathCAD). It might be combined with statistical tools for engineers, especially where predictions and forecasting of natural events is important.

- CAD applications for engineers, without the emphasis on a specific application. Depending on the needs 2D or 3D tools should be considered. It is often useful to spend some time on general computer graphics topics.
- Depending on the type of the studies it may be useful to introduce data bases, optimization or some other specialized data processing tools. It would be usually incorporated in the curricula of master courses, rather than on the basic engineering studies.

It is very important to show real life examples within each application and as much as possible to tie it with engineering lectures in the semester. This is the way to convince students of the need of IT education throughout the course of their studies.

Without elimination of good lectures in the practical part we should turn towards project-based learning, as presented in [4] , where the authors suggest, that recently students are "graduating with good knowledge of fundamental engineering science and computer literacy, but they don't know how to apply that in practice". Well, there is the need of both ends, where good convincing lecture has to be followed by the practical approach to the problems.

One cannot neglect the problem of team work as stressed in [5]. However this is the most difficult part in creating of curricula in technological universities in Poland. This problem will be further analyzed by the author.

4. Conclusions

As we can see the new curriculum of IT studies has to show general ideas in the course of the lectures, showing wide view of IT as a basic discipline. On the other hand exercises must be strongly connected with practical examples, as it is often referred to as "project-based". It is a very good idea to let the students influence introduction of constant changes to the curricula, possibly in an online contact with the teaching staff. At our University the curricula are not flexible enough and it has to be changed. We should move towards flexible course selection systems as presented in [6].

Changing the attitude of the students and presenting them with acceptable curriculum should be our aim in the constant process of the IT curricula development at technological universities.

The real problem in curricula is the position of IT education in relation with the other subjects. Basics and understanding has to be introduced possibly early (first semester), whilst practical part, after some engineering knowledge is being taught (third semester or even later). Combining these two contrary needs is a real challenge for the future.

REFERENCES

[1] Kerckhove de, D (1997) C*onnected Intelligence. The Arrival of the Web Society*, Sommerville House Publishing Books Limited, Toronto, Ontario, Canada.

[2] Kerckhove de, D (1995) *The Skin of Culture*, Sommerville House Publishing Books Limited, Toronto, Ontario, Canada.

[3] Sathepornvajana S., Watanapa B., Jungcharoensukying E. (2014) *A comparative study of student's attitudes among IT and non-IT students in Information Technology education,* International Symposium on Technology Management and Emerging Technologies (ISTMET), Bandung, Indonesia, 365-370.

[4] Mills, Julie E., and David F. Treagust, (2003) *Engineering education- Is problem-based or project-based learning the answer?*, Australasian Journal of Engineering Education 3, Australia, 2-16.

[5] Neal, P. R.; Ho, M.; Fimbres-Weihs, G.; Hussain, F.; Cinar, Y. (2012) *Project-Based Learning for First-Year Engineering Students: Design of C[O.Sub.2] Sequestration*, Australasian Journal of Engineering Education, Vol. 17, no. 2, Australia.

[6] Nakajama N., Hoshito J. (2009), *Feasibility Study of Developing a University Course Selection Support System which uses Informal Peer Suggestions*, International Journal of Emerging technologies in Learning, Vol 4, no. 3, Vienna, Austria, 29-33.