**Zeitschrift:** IABSE reports = Rapports AIPC = IVBH Berichte
**Band:** 40 (1982)

**Artikel:** Education of structural engineers in electronic data processing
**Autor:** Baldauf, Heinrich
**DOI:** https://doi.org/10.5169/seals-30911

# Education of Structural Engineers in Electronic Data Processing

Formation des ingénieurs civils dans le domaine de l'utilisation de l'ordinateur

Empfehlungen für die EDV-Ausbildung von Bauingenieuren

**Heinrich BALDAUF**
Dr.-Ing.
Bilfinger + Berger
Mannheim, Fed. Rep. of Germany

Heinrich Baldauf, born 1925 obtained his doctor's degree at the Technische Hochschule Dresden, GDR. After ten years of scientific activity at Dresden and Leipzig, GDR, he started work in 1961 in a construction firm, where he is now head of the technical-scientific computing centre.

## SUMMARY

In day-to-day professional work, people are needed who can use electronic data processing in an intelligent and critical way. An engineer must be aware that a computer is a splendid tool to help him to do his job but it must remain only a tool. These recommendations give some guidelines for education of engineers according to this philosophy.

## RESUME

Dans le contexte d'un travail quotidien, il est nécessaire de disposer de collaborateurs qui sont à même d'utiliser les procédés de calcul sur ordinateur d'une manière intelligente et critique. L'ingénieur doit se rendre compte que l'ordinateur est une aide précieuse dans la réalisation de ses tâches, sans pour cela dépasser le stade d'un outil. Ces recommandations donnent quelques directives pour une formation allant dans ce sens.

## ZUSAMMENFASSUNG

In der täglichen praktischen Arbeit werden Leute gebraucht, die die EDV in intelligenter und kritischer Weise anwenden. Ein Ingenieur soll sich bewusst sein, dass der Computer ein hervorragendes Werkzeug bei der Erfüllung seiner Aufgaben ist, aber er muss ein Werkzeug bleiben. Diese Empfehlungen geben eine Richtschnur für eine Ausbildung in diesem Sinn.

## 1. INTRODUCTION

When, about twenty years ago, the use of computers began in civil engineering, there were many limitations on their application. The computers were slow, the internal storage was small and the peripheral storage, if any, was limited. By this time, all has changed, today we can make use of very efficient computers in many ways. The size of a program is today of less importance and computer speed has increased so impressively that we can obtain results within seconds for which we needed to wait hours or days twenty years ago. In former times programs were small. Bigger problems could be handled by successive programs, but at each step the data had to be transferred, manually by cards or tapes. But these data could be changed and supplemented very easily. Today we live with complex chains of programs or systems of programs, but an interference during the run is nearly impossible, unless we run an interactive program. This means, that at the beginning of a run we have to know and to define all the data needed throughout the run. This means too, that when designing and developing a program, we must think of all the possibilities the program will have to handle in the future. Subsequent changes and extensions of program systems become more and more expensive, even if the program is divided into small modules with well defined interfaces. Testing of an extended complex program system becomes more expensive as not only the extension itself has to be tested, but we must also prove that the original program remains effective with the extension. If you have a complex program in your computer centre, you need at least one person, who is familiar with the program and who can give advice on how it should be used. He will only keep this capability if he is continually concerned with the program and can make use of it. But in your computer centre you will have not just one, but many complex programs. Today there are many finite element programs, but there is no ideal "super finite element program" which can handle all known or unknown problems. So we have to live with a multitude of programs for use in connection with shells, tunnels, drilled piles etc. In brief: compared with the situation twenty years ago, design and maintenance of software has become very expensive.
We should take into account another fact: in civil engineering we have no mass production. The cost of design and construction must be seen in relation to the cost of the entire building. The cost of construction and computation cannot be spread over hundreds or millions of identical articles. It is not a good use of a computer to carry out excessive calculations of apparently great accuracy if by doing so you can save only a fraction of the computer cost. Certainly it is no accident that the applications of finite elements are mostly reported by designers and manufacturers of machines, cars, trucks, aircraft etc.
It is with this background in mind that we must consider the following recommendations for education in data processing.

## 2. GENERAL RECOMMENDATIONS

The recommendations presented here are influenced by my own experience and by the experience and requirements of other big construction firms in the Federal Republic of Germany. I must describe them only as wishes, not demands, because the list is very long and I doubt whether all of them can be achieved through study at a technical university or similar institution. Universities are always faced with the conflict between growth of knowledge and the limited time which is available to teach it. Some aspects of education in data processing must undoubtedly be left for the time after studies at the university are completed. But the task of the universities should be seen as giving students the best possible attitude of mind towards the use of computers. In day-to-day professional work, we need people who can use electronic data processing in an intelligent and critical way. An engineer must be aware that a computer is a splendid tool to help him do his job

but it must remain only a tool and should not become an end in itself. Anybody who has ever developed programs knows how fascinating this activity is, and that there is the danger of becomming addicted to it and wasting time and effort over making a program "better and better".

## 3. THE STAGES OF EDUCATION

The education in data processing must be divided into basic and advanced education.

### 3.1 Basic education

The objective of basic education in data processing must be to make the student conscious of the possibilities and the limitations of computers. He must be put in a position to work with existing programs, to evaluate their results critically, and to write small subsidiary or supplementary programs himself.

#### 3.1.1 Learning a programming language

The student must be trained in a computer language. Since FORTRAN is the main language in technology, instruction in that language is preferable. But programming must not be confined to the beginning of his studies. It is not until his later years, when he has obtained more professional knowledge, that the student can work up to a project of practical use.

#### 3.1.2 The Structure and operation of a computer

For the utilisation of computers it is desirable to have a basic understanding of the structure of a computer, how it operates, and its capabilities. True, today computers are like cars and one can drive a car without knowing how the engine functions. When something goes wrong, however, it is an advantage to know more about a computer. The same is true for the components of an operating system, as far as is needed for the writing of individual small programs. Examples are:

The big machine in the computer centre
A terminal at the place of work
A small computer
Personal computer
Graphic display unit
Plotter
Digitizer

#### 3.1.3 Types of program operation

According to the type and size of a computer, there are different varieties of program operation. Examples are: batch operation; time-sharing with the possibility of interactive operation; remote data processing.

#### 3.1.4 The application of computers in civil engineering

The application of computers is tending to increase in all areas of a construction company. The number of civil engineers who are concerned with the development of new programs will remain small, but all engineers will come into contact with computers in one way or another. They will have to work with programs which others have developed. This involves the ability to read a

program description (in the television age this ability cannot be assumed!).
The user must learn to make judgements on:
What can the program accomplish?
What can the program not accomplish?
He must remain critical and ask:
How can I check the working of a program?
(coordinate transformation, rotation, reflection)
How can I check the results in a particular case?
(equilibrium conditions on a single element or global cross-sections, compatability conditions between internal forces and displacements.)

### 3.1.5 Finite elements

Finite element methods are being employed more and more frequently for computations; these require a specially critical attitude on the part of the user. An engineer should become familiar with the fundamentals and the limitations of certain classes of finite elements but in his basic education he does not need to know how to design new elements. Much more important, for the user, are questions regarding the ways in which accuracy depends on the network structure, the number of elements, and so on. Particularly important is the question of controlling a finite element computation, for the worst control is the attitude "You don't need to control finite element calculations, you can just believe in them".

### 3.1.6 The economics of computer applications

As was mentioned in the introduction, the economics of computation are highly significant in engineering practice. In universities and similar institutions, one encounters mostly very large, fast computers, and computing costs are relatively unimportant. In practical life, an engineer must be prepared to recognise that unlimited computer capacity may not be available and that a price-tag must be attached to the use of computer time. He must also learn to see the cost of computer use in relation to the overall cost of design, construction and calculation.
The user must evaluate:
Does the aim of the calculation (preliminary calculation, execution, control) justify the application of an expensive program?
Does the accuracy of the initial data justify the use of a "more accurate" calculation procedure?
Will high computation costs be compensated by higher savings?

## 3.2 More advanced education

The higher stages of education should put the engineer in a position to develop programs independently, from the analysis of the problem to the actual programming. He must be able to make himself familiar with existing programs, to eliminate errors from them, and to carry out modifications and extensions to them. We must strengthen and develop the studies which we have mentioned at the fundamental level. The following areas of knowledge are particularly important:

### 3.2.1 Mathematical methods

The most important matter here is to recognise the conditions under which existing mathematical methods can be used. A civil engineer does not have to solve mathematical problems, but has to use mathematical techniques correctly in solving his engineering problems:

Matrix caclulus (including the solution of small or large systems of linear equations)
Iterative procedures (criteria for terminating an iteration, from various points of view)
Optimization techniques
Numerical solution of differential equations

### 3.2.2 More advanced programming

The difficulties in constructing a program increase with the degree of complexity of that program. One has to consider all the aspects of what is now called Software Engineering.

Structured programming
Structure of data flow
Breaking down a large program into small modules
Incorporation of modules into other programs

Today, we can only think about the coupling of programming systems after satisfactory specification of interfaces for data.
Consequently, it is desirable to give students also an introduction to data management and data banks. The increasing practical importance of graphical representation of data must be brought home students by means of an introduction to graphical software and CAD systems. Furthermore, increasing use in applications of interactive processing requires that a programmer should be familiar with this technique.
The quality of a program depends, moreover, on its compatibility and portability with respect to interchange of hardware and operating system software. Hence our aim must be to use the language only in a standard form. "Clever" programming may be great fun for programmer but it can have disastrous consequences in routine usage.

### 3.2.3 Documentation

A piece of software is only as good as its documentation. Since the expense of documentation is often as high as that of developing the program itself, documentation is often neglected. In universities, programs are developed under the time pressure of writing the Diploma thesis, and consequently the process of documentation does not receive proper consideration.

### 3.2.4 Program development

Misunderstanding and confusion between programmers and their customers is mostly caused by insufficient and imprecise program specification. The student must therefore be made to realize that good programming depends on a continuing dialogue between programmers and customers.

## 4. INFORMATION PROCESSING AND CIVIL ENGINEERING

The recruitment of mathematicians into the construction industry has often led to disappointment on both sides, caused by differences in professional attitude. The engineer is wanting results, often under pressure of time. If the results are consistent with his previous experience, and can be verified by an independent calculation, then he will accept them. By contrast the mathematician sees his task as that of theoretically guaranteeing the existence of a solution. The result is a communication gap. Some of the mathematician's procedures are familiar to the engineer, but under other names. It often takes a long time to get on the same wavelength.

If a computer scientist is to be successfully active in the construction industry, he has to be very clear that to an engineer it is the end result of the program which matters most. He is not free to regard the program as a work of art, which regrettably must be contaminated by contact with the outside world through input and output. Being introduced to actual engineering work should help to give the programmer an appreciation of the opposite point of view.

## 5. RESEARCH AND FINITE ELEMENTS

Young engineers often come out of university with the impression that the only point which needs to be justified is the actual use of a finite element program. Then they are very disappointed when they hear that in daily practice finite element applications form only a fraction of all applications. I regard this kind of education as inadequate. A finite element method is still just an approximation procedure, even though it is a very good one. The use of finite elements is necessary where higher accuracy is required and there is no other computational procedure available. But it is wrong to use a finite element program on every conceivable problem, just on principle. This applies particularly in cases where we have no guarantee of the accuracy of the input data to a calculation process. In spite of that, such results are often called "exact". In the situation where initial data may have a probabilistic distribution, we have to make an extensive study of the parameters before we can devise a simple computational procedure, for example, in construction of foundations or of tunnels.

## 6. CONCLUSION

Our list of "wants" is extensive and long. For education, the important thing is to develop a critical attitude towards the use of computers. The computer will in future be employed more and more as an efficient tool in the construction industry. The objective, however, must not be "use the computer whenever you can, no matter how much it costs" but "use the computer whenever it is professionally necessary and justified".