

Technology of development of computerized training courses

M.M. Bezhanova, G.A. Kvashnin, L.A. Moskvina

The basic principles and tools of the technology, intended for the development of the computerized training environments, in particular for computerization of university courses are presented. The proposed technology is based on the use of the processor Word for Windows and other environments, such as Turbo Pascal, Delphi, etc.

1. Introduction

The computerized training course (CTC) is one of the form of training programs [1] and represents a complex of program and information means, intended for studying a scientific discipline, development of skills of behaviour in certain circumstances, etc. The technology considered in the present work is aimed at the development of the general approach to creation of such courses for rather wide spectrum of fields of knowledge.

The offered approach permits one to realize on the computer most adequately, for example, the principles of university education:

- plunging into problem-oriented environment of a discipline with the “tools” inherent to it, i.e., dictionaries, illustrations, simulated processes, etc.
- gaining the extensive profound knowledge on various scientific disciplines and opportunities of their independent study,
- checking the quality of the gained knowledge when fulfilling the educational tasks,
- using the gained knowledge when solving practical problems of various complexity.

The proposed means and techniques permit the implementor of the course to define the necessary manner, the character of a dialogue the CTC with the user. On the basis of the specified technology a prototype of the course “Methods of Monte Carlo” has been created. The course “Practical Programming” is being created at present.

2. Supporting the principle of the multi-level training

CTC represents a package (a set) of interacting documents and files. This package has several inputs, each being oriented to gaining the knowledge of a specific kind and character, and corresponds to different levels of students' preparedness. Many documents are made up as directories in different branches of knowledge and are intended for different purposes of education. There are:

- dictionaries of concepts and terms of general importance (for example, concepts of computer science),
- dictionaries of the terms from narrow, particular application areas (for example, concrete programming system),
- the lists of problems (or programs) of different types (for example, programs of classic algorithms or programs of modeling of different data structures),
- additional opportunities and details of realization, etc.

A trainee turns to the documents – directories already having, as a rule, an idea of the field of knowledge under study. Therefore, one of the main principles of our approach to the computer training course consists in that gaining such a general knowledge about a subject of study should be primary in the course of training and most carefully prepared. In any scientific educational discipline such basic knowledge is presented, as a rule, as a hierarchically ordered information, usually textual. By virtue of it each CTC includes the main document, containing this information, and the input into it is considered as the main input – a starting point of a training course. All other inputs (into documents – directories) are considered as additional, auxiliary. Their purpose is in obtaining a particular kind of reference information.

3. Support of the prototyping principle

One of the main principles of the considered technology of the computer aided training is program prototyping. A syntactically and semantically correct program is referred to as a program-prototype or simply a prototype. Its purpose is a basis, an example of any typical program object: a module of a particular kind, structure of management, types of data, etc. A prototype model of an object is the description of informational or managing structure of an object. The prototype model is usually oriented to a particular programming language and can serve an example of presentation of fragments of the user's programs in this language. A representation

recommended for a given typical object, its realization in a certain programming language are described in a prototype model of an object (as differed from its specification). In this case it is essential to know how, by what linguistic means, to design, to simulate this object. The value of the concept of a prototype is in that it concentrates the trainee's attention on the important parts of the object structure.

In a way the presented technology can be considered as a technology of "training on examples". Teaching to design programs is conducted by commenting examples (models): they should be made up to be visually perceived and stored. The prototype models of programs in the considered technology are an important reference means of teaching the methods and techniques of programming.

The greatest interest and importance the prototyping technique acquires for complex objects, whose description requires extensive and profound knowledge of the programming language and features of its realization for gaining the maximal efficiency, reliability, flexibility and other properties of a "good" style of programming.

In the considered technology the prototyping principle is the basis in the creation of collections of different kinds of problems (programs). The same technique can be considered as a way of development of the tool environment: a supplementable collection of models (prototypes) of specialized tools is created. By tuning or modifying them (with the use of the offered instructions and recommendations) a collection of tools, suitable for a new CTC can be easily obtained. Such an approach to the development of typical program tools has been realized and proved to be correct in the support system TRAP [2].

4. Selection of environment for development and functioning of CTC

A special tool environment has been created for the program support of the above-presented technology intended for the development of computerized training courses. It is based on the use of the means, provided by wide applications of the system Windows, in particular, by the textual processor Word for Windows [3-4].

The trainee's work in the environment, provided by the computerized course looks like the work with the interactive program. The dialogue with such a program takes place by initiation of specialized menu commands choosing an item or pressing a button of specialized panels, setting the required information in the dialogue windows.

By indication to a phrase or a picture in the text, specially marked by colour, italics, font it is possible to go to necessary fragments of the course,

to come back to the earlier looked-through fragments, to start execution of programs, to address to various kinds of directories. The environment, supporting such a style of work, provides:

- well formatted visual texts, containing various signs, formulas, pictures, schemes,
- hierarchical and interrelated presentation of information, ensuring a convenient viewing of its different fragments and storing of such a viewing,
- running programs and films visually simulating the functioning of algorithms and processes under study.

5. Means of creation of problem-oriented environments

The tool environments are one of the widespread types of extendable, modified and re-used program environments. The tool environment is usually understood as software environment, created by a collection of program tools, developed on the basis of general principles, and by a certain basic set of mutually complementing means. The removal of any tool from a collection does not result in destruction of the environment, but only reduces its possibilities.

The tool environments can borrow a number of their components from other environments, and in their re-use they are easier adapted to new requirements due to the possibility to replace (redefine) their components. As tool environments there are designed environments oriented to non-unique, mass use, for example, educational and training programs.

In the considered tool environment the textual processor **Word for Windows (WinWord)** is used as a basic means. The **WinWord** possesses perfect opportunities of formatting, different forms of presentation of a document, which are effective for the fulfilment of various works in its creation, storage and updating. It is also very convenient for mathematical texts, containing a plenty of complex and bulky formulas. The study of **WinWord** has shown, that it contains a number of means, which are seldom used at preparation of textual documents, but can find effective application when designing applications [5]. To such means we refer:

- the means of creation of the problem-oriented environments,
- the means of creation of hypertextual presentation of information,
- the means of creation of a convenient and diverse user's interface.

A specialized problem-oriented context (environment) can be defined for each document or certain package of documents. The environment support

of the document in **WinWord** is a template. A template of the document is a model for manufacturing a new or a typical document. This template can include styles, macros, menus, fast keys, panels, etc., i.e., all those objects of non-textual character which should be specified or defined, which are used in the work with a document. It may be necessary to use the same document (similarly to a program) in a few environments. It can be done, by having defined for a document its own template and having specified the use of several already existing templates.

Usually a specialized template is created for a certain set, a collection of the documents, functioning in cooperation in one environment. When opening the document the access to all objects of its templates is provided. Moreover, at the same moment a number of actions (initialization), described in the automatically running system macros are executed, which can be redefined and placed in a template. Sometimes it is necessary to change the environment of a document (an interface in particular) while working with it. This can be done by macros described in a template and called at execution of the document.

The problem-oriented user's interface of the document is defined by a set of specialized menus, panels of tools, dialogue windows. It is possible both to modify the standard interface means of the document (provided by **WinWord**), and to design absolutely new means, necessary only for a given group of users. The main basic means, with whose help such an effect is attained, is the creation of macros and their comparison to interface objects (menu commands, buttons of panels, elements of the dialogue).

6. Hypertextual presentation of CTC information

A document can be constructed as a hypertext [6]. It is attained by building in it of such elements as layings, fields, objects – for the description of its structure and linkage with such components of a hypertext, as formulas, drawings, tables, executed macros and programs, fragments of databases, etc. The book marks are one of the most useful elements in **WinWord**. It serves as a marker of that place in the document, to which one may need to return not once.

The book marks are used in instructions of fields. A field in **WinWord** is a means of inclusion in the document of its various elements. This is a specific "capsule" with a set of instructions, which define formatting and insertion of a text or an image or execution of specified programs. A great number of types of fields are intended for formatting of the document, insertion in it of drawings, texts, formulas, etc. However types of fields, making the document active, executable and determining its hypertextual structure, are

the field of transition to a specified place in the document (book marks) and the field of a call of a macros. An important hypertextual element of the document is an object. With its help the information, created by another application of **Windows**, supporting the mechanism OLE (Object Linking and Embedding) can be imported to the document **WinWord**.

In addition to objects in the document it is possible to introduce the packed objects (packages) – pictograms of files, including the executed ones, which can be obtained from the application – the source OLE. It is a very powerful mechanism of connections with components of a hypertext, enabling one to execute practically any programs by activation of a pictogram of a packed object. Activation of fields of the above-mentioned types (execution of actions represented by them) is executed at the user's will when viewing the document. Therefore as differed from a static information form (a text, a drawing, etc.) a document is an active form of representation of information. It can be considered as a set of components to be executed. As differed from a program, executed by a processor, following a certain sequence of commands, the document is executed from events, occurring when it is opened, looked through, as well as the user's commands, determined in the dialogue mode. Another dynamic feature of the document is the possibility of changing its interface in the process of viewing (execution).

Each particular training course is created as application, i.e., a package of documents with templates, describing the problem-oriented environment, interface objects. A package of documents represents a uniform hypertext: viewing with transition in accordance with every possible reference both inside the document, and between the documents is possible. In addition to the package of documents, created with help **WinWord**, the computerized course can contain files, created in other environments (**Delphi**, **Painbrush**, **Turbo Pascal**, etc.).

A tool environment for functioning of a particular CTC consists of the basic processor **WinWord**, tool means, with the use of components, developed in other environments, and a collection of tools, making up the problem-oriented environment of CTC (templates, macros, panels, menus). A number of tools of the collection is common for all CTC, but some of them reflect a specific character of a particular course.

The main objective of the computerized training course is the help to a trainee in active development of a particular problem area, which includes a goal-directed study of theory (with the help of various forms of its representation) gaining the skills in the use of the knowledge obtained when solving problems. A computerized training course contains, as a rule, a variety of programs, realizing algorithms, methods and structures of the given subject area. Using these programs a trainee can learn to build models of investigated processes and phenomena, to design algorithms of solution by choosing a more effective representation, to investigate algorithms, to de-

fine their characteristics. These programs also make it possible to provide several levels of studying the educational material:

- demonstration of execution of an algorithm with fixed initial data,
- setting various initial data and studying their influence on the work of algorithm functioning and results of the solution,
- obtaining other variants of the program by its updating,
- creation of programs of the solution of related problems using the given one as a prototype.

References

- [1] M.M. Bezhanova, I.V. Pottosin, *Computer Software: Tools and Training Means*, Novosibirsk State University, Novosibirsk, 1996 (in Russian).
- [2] M.M. Bezhanova, *Principles and means of development of applied program systems*, Computing Center, SO AN SSSR, Novosibirsk, 1987 (in Russian).
- [3] M.M. Bezhanova, G.A. Kvashnin, L.A. Moskvina, *Environment of development of electronic manuals*, Trudy VTs SO RAN, Seria Vychislitel'naya Matematika, vol. 5, Novosibirsk, 1996, 20–28 (in Russian).
- [4] M.M. Bezhanova, G.A. Kvashnin, L.A. Moskvina, *Computerized training courses in the environment Windows*, Preprint, Computing Center SO RAN, No. 1088, 1997 (in Russian).
- [5] M.M. Bezhanova, *The tool environments for the creation of hypertextual interactive applications*, Preprint, Computing Center SO RAN, No. 1084, 1997 (in Russian).
- [6] M.M. Bezchanova, *The review of means hypertextual technology*, Trudy VTs SO RAN, Seria Vychislitel'naya Matematika, vol. 5, Novosibirsk, 1996, 12–19 (in Russian).