[9]   Redondo M., C. Bravo, M. Marcelino, A. J. Mendes. Tools for Programming Learning an Approach to Provide a Social Perspective Using Collaborative Planning of Design. *IADIS International Conference e-Society 2004,* Avila, Spain, 2004, July.

[10] Stasko, J., Animating algorithms with XTANGO, SIGACT News, Vol. 23, No.2, Spring, 1992, pp.67-71.

[11] Zheliazkova I. I., An intelligent system for teaching and learning algorithms, Int. J. Computers & Education, Vol. 24, No. 2, 1995, pp. 117-125.

[12] Zheliazkova I. I., Andreeva M. H., An Intelligent Multimedia Environment for Knowledge Testing, E-learning and the Knowledge Society, Gent & Brussels, Belgium, 6-8 September 2004, pp. 3.13.1-3.13.24

[13] Zheliazkova I. I., Kolev R., Multi-parametrical Analysis of Test Results, Computer Science'2005, Halkidiki, Greece.

[14] Zheliazkova I. I., G. Atanasova. A Visual Language for Algorithm Knowledge Representation. International Conference on Computer Systems and Technologies (e-Learning), Rousse, Bulgaria, 2004, 17-18 June, IV.24-1 – IV.24-6

[15] Zheliazkova I. I., Atanasova G. Computer-aided teaching and learning algorithms (A survey), Proceedings of the International Conference "Innovation in Education for Electrical and Information Engineering", 27th-29th May, Sofia, pp. 49-58.

[16] Zeliazkova, I.I., Andreeva M., Grozdanova E., Experimental Study of the Computer-based Knowledge Testing in the Primary School, Conference of RU'2004, section "Mathematics, Informatics and Physics", pp.95-100

[17] Zeliazkova, I.I., Andreeva M., Nikolova M., Experimental Study of the Computer-based Knowledge Testing in the High School, Conference of RU'2004, section "Mathematics, Informatics and Physics", pp.89-94

[18] Zeliazkova, I.I., Andreeva M., Experimental Study of the Computer-based Knowledge Testing, 31.10-01.11.2003, Rousse, Bulgaria, Conference of RU'2002, pp.48-52

Authors' Information

Irina Zheliazkova – Associate Professor; Rousse University, str. "Studentska" 8, Russe-7017, Bulgaria; e-mail: irina@ecs.ru.acad.bg

Magdalena Andreeva – Senior Assistant; Rousse University, str. "Studentska" 8, Russe-7017, Bulgaria; e-mail: magie@ami.ru.acad.bg

Rumen Kolev – PhD Student, Rousse University, str. "Studentska" 8, Russe-7017, Bulgaria; e-mail: rkolev@ecs.ru.acad.bg

DEVELOPING COLLECTIVE TEACHING COMPUTER SOFTWARE FOR THE COURSE "DECISION THEORY"

Konstantin Berezovskiy, Olexy Voloshyn, Igor Drozdov

Abstract: *Improvement of training students using modern information technologies, like collective developing teaching computer software, is discussed. Organizational, technical, technological advices are given. Experience of using information technologies in educational course "Decision Theory" is described..*

Keywords: *Electronic learning, information technologies, teaching software, collective developing, Decision Theory.*

Introduction

At creation of normative documents for higher educational establishments of Ukraine in 1992-1994, after finding of independence Ukraine on initiative of one of authors, member of scientifically-methodical commission of Department of education of Ukraine on the Applied Mathematics, in the curricula of direction «Applied mathematics», «Decision Theory» course was included. Experience of the best educational establishments of the world such, as Cambridge, Oxford, Californian University (Berkeley), Massachusetts Institute of Technology etc. was taken into account. About importance of similar course testifies a fact, that for the last decade for results got

in the theories of decision-making and its use in many areas of human activity(above all things, in economy), two Nobel awards(1994 - J. Nash, J Harsanyi, Zelten; 2005 - Auman) are won.

Adhering to classification of G. Simone and A. Newell [Voloshin, 2006], direction-finding problems behave to the semistructured problems. Then the typical problems of operations research belong to well structured, problems of artificial intelligence to unstructured. Thus, in the view of curriculum of direction «Applied mathematics», a "Decision Theory" is an intermediate between normative courses "Operations Research" («Methods Of Optimizations») and «Artificial Intelligence» («Knowledgebases Planning»).

The methodical providing of the course is «Decision Theory»

On the faculty of Cybernetics of the Kyiv Taras Shevchenko National University educational courses "Decision Theory" are on all departments(informatics, social informatics, applied mathematics) on the third year in a volume of 36 hours of lectures and 18 hours of practical or laboratory employments(one semester, one lecture in a week and one practical or laboratory employment in two weeks). During ten years lecturers from a lot of higher educational establishments of Ukraine (which prepare specialists to direction «Applied Mathematics») applied on the faculty of Cybernetics with a request to recommend a list of themes and literature on this course, as school-book (not to mention about textbooks), in which full basic sections would be reflected on the "Decision theory", in world practice are absent. The ten years experience of teaching courses "Decision Theory" on the faculty of of the Kyiv Taras Shevchenko National University professor Voloshin A.F. and reader Mashenko S.O. is generalized in an educational textbook «Decision Theory» [Voloshin, 2006] which in the moment of writing of these theses is in a print and it will be presented at conference MeL2006("Modern (e-)Learning", on July, 1-5 2006, Bulgaria in Varna). In a textbook on the view of its authors, the basic sections of decision-making theory are presented:

1. Philosophical and mathematical bases.
2. Theory of utility.
3. Expert procedures.
4. Decision-making in the conditions of vagueness and risk.
5. Multicriterion optimization.
6. Decision-making in the conditions of a conflict.
7. Co-operative decision-making.
8. Decision-making in the conditions of unclear information.

First five years teaching the course, professor Voloshin A.F. was accompanied practical employments, where on examples the basic moments of theory were examined, algorithms were illustrated on numeral exercises.

Understanding came later, that in teaching of course «Decision Theory» main things are not concrete algorithms(which mostly have heuristic character), but the basic ideas. Algorithms, and yet better their programmatic realization, serve for an analysis, illustration and understanding of basic principles of decision-making theory.

First two semesters of conducting of laboratory employments, the tasks of students consisted in development of the programmatic modules, realizing concrete algorithms [Voloshin, 2006], without the concordance of programming, language, environment, etc.

Two semesters went away on the attempt of creation of programmatic modules library, incorporated within the themes (for example, multicriterion optimization).

Collective software product

In the last 2 semesters, there have been several attempts (that was partially implemented) of the creation of "collective software product that will have methodical and learning features". The goal was to solve common problem, which is shown on the scheme in figure 1.

To create a software product that will fulfill this scheme, there is a need of solving several organizational problems. We need to take into account student's different level of understanding of mathematics and programming, time constrains for the elaboration of the product (18 hours of auditorium lectures and not more than 18 hours of independent work). Generally, in the course of the semester students need to do 3 laboratory works (first week – defining the task, third – consultations for the solving a problem, fifth- completing the

assignment). Two students were selected to create an interface (it would be better if that would be "volunteers", that are responsible, competent, that also have authority over other students). With them, the preliminary plan was done (this was actually hard to do, since at the begging of the course students do not now it's content).

The system is supplied with inquiry function, that contains description of separate modules and algorithms that allow people to use the product even with minimal knowledge of this course «Theory of decision-making». We think, that the system is most useful in distance learning, and can be used by students of non-mathematical specializations and distance-learning students, that study this course.
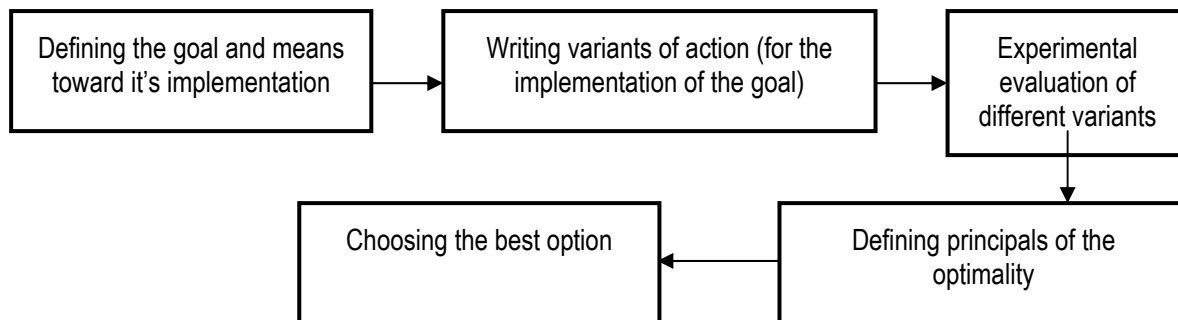
```
┌─────────────────────┐     ┌─────────────────────┐     ┌─────────────────────┐
│ Defining the goal    │ ──▶ │ Writing variants of  │ ──▶ │  Experimental        │
│ and means toward     │     │ action (for the      │     │  evaluation of       │
│ it's implementation  │     │ implementation of    │     │  different variants  │
│                      │     │ the goal)            │     │                      │
└─────────────────────┘     └─────────────────────┘     └─────────────────────┘
                                                                    │
                                                                    ▼
       ┌─────────────────────┐                         ┌─────────────────────┐
       │ Choosing the best    │ ◀────────────────────── │ Defining principals  │
       │ option               │                         │ of the optimality    │
       └─────────────────────┘                         └─────────────────────┘
```

Figure 1

Some recommendations for the makers of the collective software products

Our experience in making such training products give us the possibility to come up with such recommendations:

1. Choosing means and software product's development environment, due to such reasons as: convenience, platform independent, potential user's software abilities software;
2. There should be installed technique principals of programming with the goal of using the code again and it's support by any creator, that is in the team;
3. Structures of software products should be planned in a way that integration of parts, that were created by separate creators or the group, would be max lightened, into one whole;
4. Set a time limit, when each participant would do it's part (module);
5. The coordinator of the project and its assistances should periodically familiarize themselves with the results of work, done by separate creators. This from one hand would stimulate working group to complete their assignments on time, and from another, will help to solve problems that will arouse while projecting;
6. Every participant should write technical documentation and information for users in the context of their created module. There should be an opportunity for search inquiries of the needed material in the context of the project as a whole;
7. Modules testing should be organized in a way, that one creator would be checking other creator's work, that should be done up to the integration of separate modules into the whole product;
8. Software product should be tested by all the product creators;
9. There should be a possibility of communication between creators and users, that will help to improve the product, but only in case of it's further support and development ;

Comments to recommendations

The creation of the collective software product can be effectively realized in the context of module-ratings system knowledge evaluation, that assumes time limits for the acceptance and evaluation of intermediate results (program module) and giving penalty points for late handing works. If it's possible, some student's software modules development should be duplicated. Authors suggest that while developing a software product, it's better to follow a principle " better less, but working" rather than "better more, will complete later" (which is often the case while developing industry related software product; after getting the final grade on the exam – there is no "later").

According to the testimony of one author, first of all "creation of collective software product (even educational) is much harder that it seemed at fist glance", and secondly "none teaches how to do this in the university department", although modern software products are 90% done collectively"

Here it needs to be added that the course "Theory of decision-making" is being taught in the 6th semester, and the specialization has clearly a software orientation.

Authors want to attach to books several disks that will have 2 training systems:

1.   A set of software modules, that will realize separate algorithms;

2.   A system of mutually connected software modules that will realize the fulfillment of individual topics.

The second system's fragment is shown on the scheme, shown on figure 2.
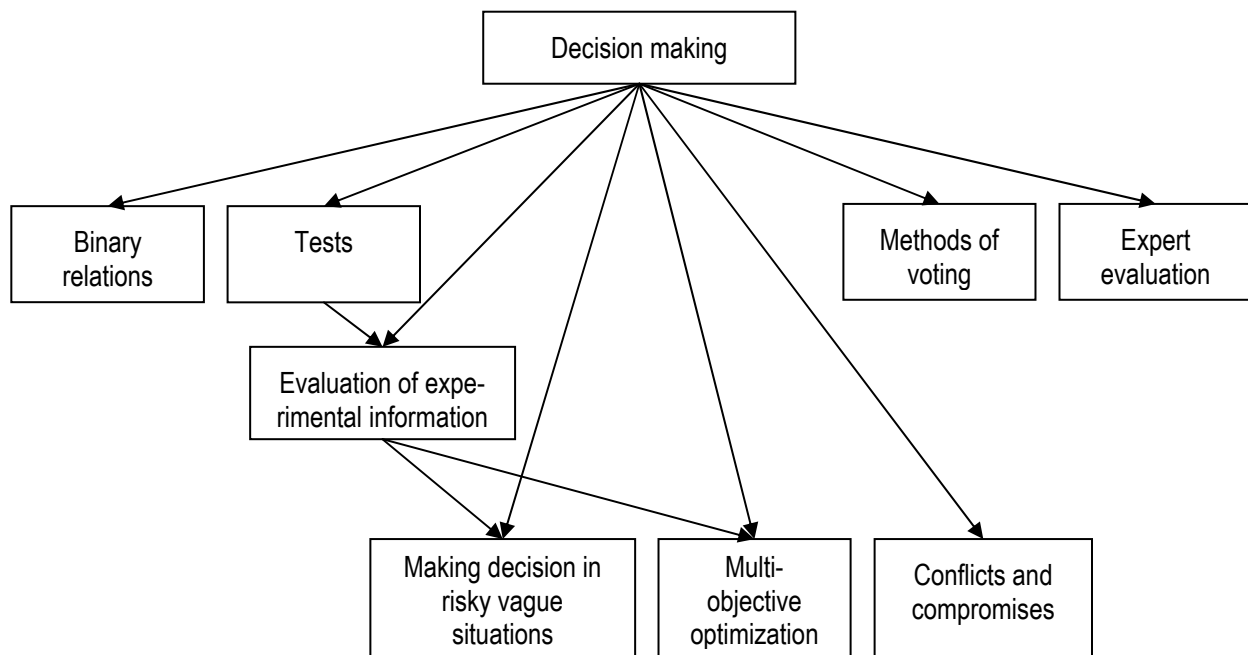

Figure 2

Conclusion

The creation of such collective software products that have methodical and learning features, not only improve the quality of computer science students preparation, but also gives an opportunity to create studying systems, that improve the quality of teaching mathematical science courses and to use new IT technologies in the distance learning education.

Bibliography

[Voloshyn, 2006] Voloshyn A. F., Mashenko S. O. Theory of decision-making. Manual. – Kiev: Publishing center «Kiev University», 2006. – 304 p. .(in Ukrainian).

[Voloshyn, 2001] Voloshyn A. F., Mashenko S. O. Methodical recommendations to carry out practical and laboratory task on the decision-making theory. Kiev: Publishing center «Kiev University», 2006. 46 p. (in Ukrainian)

Authors' Information

Konstantin Anatolievich Berezovskiy– student of the Taras Shevchenko Kiev National University, cybernetics department, e-mail: kos_berezovskiy@mail.ru

Aleksey Fedorovich Voloshin – professor of cybernetics department at the Taras Shevchenko Kiev National University, e-mail: ovoloshin@unicyb.kiev.ua

Igor Vadimovich Drozdov - student of the Taras Shevchenko Kiev National University, cybernetics department, e-mail: div@univ.kiev.ua