

IMPLEMENTING ITS (INTELLIGENT TUTORING SYSTEMS) FOR JAVA PROGRAMMING E-LEARNING

NEBOJŠA GAVRILOVIĆ

Belgrade Metropolitan University, nebojsa.gavrilovic@metropolitan.ac.rs

SLOBODAN JOVANOVIĆ

Belgrade Metropolitan University, slobodan.jovanovic@metropolitan.ac.rs

Abstract: *This paper analyses intelligent tutoring systems (ITS) for Java programming e-learning. An overview of existing ITS systems for Java programming language e-learning is presented. Their capabilities and possible applications are analysed. Then, considering how these ITS are actually used around the world in practice, in various universities or in other organisations (at universities, schools, selflearning or various programming courses) is described, and what are the main implementing problems of implementing ITS from technological point of view, and also from e-learning point of view (so, important problems detected from students and teachers and developers). Finally, some experience in developing of an ITS for Java at the Belgrade Metropolitan University is presented, and main capabilities of this ITS.*

Keywords: *E-Learning, Intelligent tutoring systems, ITS, Java,*

1. INTRODUCTION

Intelligent tutoring systems (ITS) have many applications for the purpose of learning of programming languages [1-5]. They simulate laboratories for programming languages training and practical exercising. ITS are typically based on short theoretical material and many practical examples which are offered to students to be solved by themselves. These examples can be of various types, eg. examples which need to be solved partially or completely, or solutions with errors can be presented and students are required to discover errors. Also, the teacher role within ITS has various options, for example to create various learning and testing scenarios for various students.

The main idea and purpose of ITS is to substitute/minimise the human teacher role in the process of learning. Namely, ITS offer students first to read and study theory, and after that to solve many practical problems without human teacher's help. However, ITS include an emulation of a human teacher ie. a substitute of a human teacher. In other words, ITS include a module which is capable to partially replicate a human teacher. Namely, a human teacher logic is programmed within ITS.

This paper presents an overview of existing ITS for Java programming language e-learning purposes. The main capabilities and features of these ITS are analysed. Also, some experience in developing an ITS for Java at the Belgrade Metropolitan University (BMU) is described. This BMU Java ITS enables students to prepare themselves for Java exams/tests, and to test their knowledge and readiness for Java exams.

2. OVERVIEW OF ITS FOR JAVA

ITS for Java are typically developed as Web based systems [1]. They consider various Java objects and produce exercises and problems to be solved. During this process ITS monitors the student progress. ITS checks the student knowledge and on the basis of this it gives problems which need to be solved by the student. It includes a module which emulates teacher decisions, eg. gives problems solutions to the student, and controls the student activity. Image. 1 shows the structure of ITS. And Image.2 presents the main user menu of ITS. Here we will describe two ITS for Java: (1)JO Tutor and (2) SimStudent.

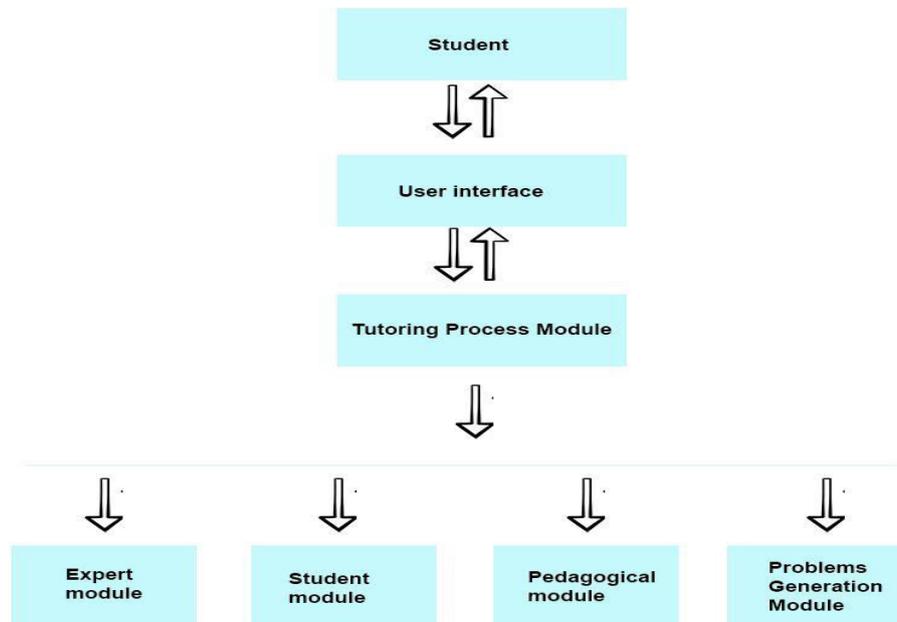


Image 1: Modules of ITS

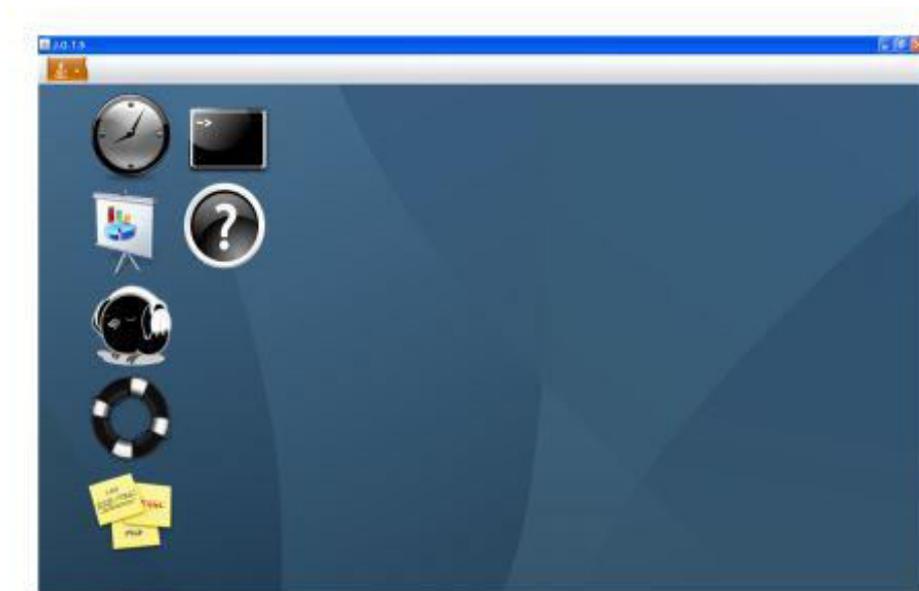


Image 2: The main user menu of ITS for Learning Java Objects

JO-Tutor ITS:

This ITS for learning Java objects (“JO-Tutor”) has a collection of exercises stored in a database [1].The system has been proved to improve student learning productivity by 10%. Students learn step by step, by learning simple parts first and complex parts later. The system automatically produces exercises and during the process of exercising it uses feedback information from the student performance. JO-Tutor is designed to support traditional learning process which uses Java books and human instructors. Also it can be used in computer laboratories for Java exercises. Also, this ITS can be used for Java home work, and can be placed on a Web server. Java programming is a complex subject for many students, and JO Tutor can help such students.

This ITS has a capability of evaluating and monitoring of student learning process. It includes a module for these purposes, which collects necessary information and produces feedback information about student learning results. For every student and every group of questions there is a evaluation result. These results can be viewed at any time, and by viewing this data we can observe how much any student is successful in any area.

User interface of JO-Tutor: After logging in, the main form window enables generating questions. Questions can be of the following types: multiple choice, true/false, one answer. Also, a student can choose either to read lessons, or to choose a topic for studying or to ask for some help .

SimStudent ITS :

SimStudent ITS e-learning system enables learning of several programming languages (not just Java) and other subjects also [2]. It is used at various universities for various subjects. SimStudent enables various functions: testing students, or just presenting lessons. For teachers, there is an author window, where lessons and tests and learning scenarios can be created/entered. This system is developed to be used as an additional support for traditional learning, ie. it is used as an additional learning tool for programmers. The purpose of the system is to improve capabilities of programmers. Image. 3 illustrates the user interface of this system.

The user interface simulates a virtual classroom. The system gives a practical programming problem which needs to be solved by a student. Before that, a solved problem example is presented to the student. Then, the student can either to solve a problem or just to describe a

potential solution. Another way of studying is requiring from the student to describe in details a complex solved problem (a case study example), and to identify concepts used in the presented solution.

The system monitors the time which the student spent for studying a certain problem, and on the basis of that evaluates is it necessary or not to add more questions/examples to the student from the same field.

Also, the system evaluates average marks for every student in ITS final report. As a result, we have personalized scenarios/strategies of learning for every student, by using student learning results in various stages of learning. Finally, the system enables human teachers to check continuously the student learning results and if necessary to add more examples in a concrete learning scenario.

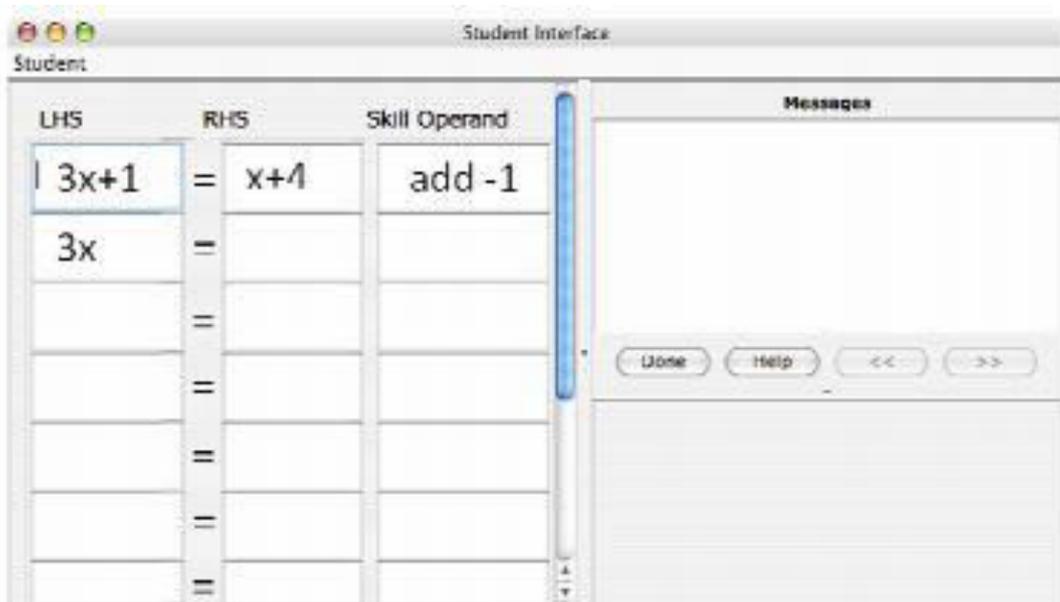


Image 3: Illustration of user interface of SimStudent ITS

3. USING JAVA ITS

ITS enable teachers to work with students and to monitor learning process of students. This is the basic philosophy of ITS. The ITS is designed to be an online laboratory

where students can solve various problems without teachers. Before starting using ITS, it is necessary to study a manuel which includes all necessary instructions.

Learning process include several parts: concepts, examples, tasks. Examples are needed to help students to be able to do his tasks. These examples can be in the form of solved exercises or theoretical hints. Tasks are in the form of a question/problem which the student needs to solve. Also, a solved problem with errors or with missing parts, which needs to be corrected. After this, the

student can proceeds with learning process ie. to enter a next stage of learning scenario.

The students are evaluated from three points of view:

- Concepts
- Examples
- Tasks

In accordance with this, there is of course three kind of learning materials for students.

The students can work individually or in a group, and in both cases using ITS has been proved as very useful. First steps in learning process are very important for a student, because in this phase a learning scenario is created by ITS.

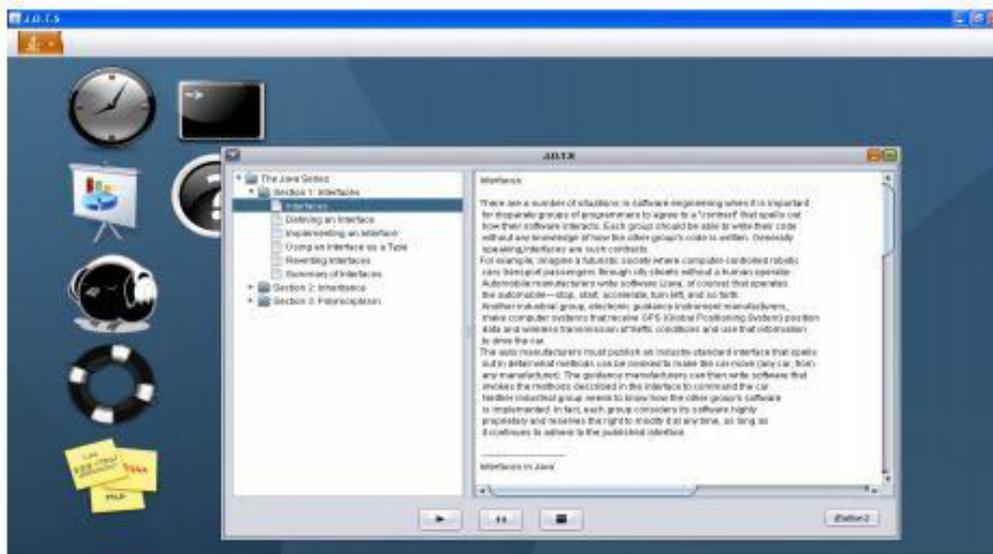


Image 4: Reading a lesson and a personalized learning sequence

ITS are capable of evaluating student knowledge and on the basis of that evaluation to choose which exercises&problems needs to be given to students. When a student is finished with an exercise (solving a problem), after that he/she can see the solution of the problem and compare it with its own solution. After that, the student can contact his/her teacher for additional help, if necessary.

4. BMU EXPERIENCE IN DEVELOPING JAVA ITS

An ITS for Java e-learning has been developed at the Belgrade Metropolitan University: BMU Java ITS (see Image. 5). Here we briefly describe the concept and capabilities of this system. The system implements artificial intelligence, ie. the logic and knowledge of a Java teacher is memorized within this BMU Java ITS. This system evaluates continuously the student knowledge and ambitions, and as a result of that it is continuously creating the learning scenario (a map of learning) which is personalized for every student. The human teacher supervises using ITS, he/she can check results of using it by students and if necessary he/she can contact students. So, the human teacher logic is stored in the system in form of a set of simple rules, and after that the ITS uses feedback information from the student and decides in every step of learning process what is the best way forward. There is a knowledge base with exercise problems and solutions of these problems and corresponding theory lessons, and these are given to a student using memorized human teacher logic (logic rules). So, the learning process generated by ITS is personalized for every student, depending of his speed of learning (some students learn quickly and some not), and ambitions (some students are interested in higher marks and some not). This personalized learning process is generated by using memorized logic rules of a human teacher and the knowledge base.

This ITS is developed to be used at universities for preparing Java exams, and this ITS is used as a substitute for human Java teacher. This system offers a whole variety of options during personalized learning process. The student can choose any field of Java, and to proceed to learn theory and perform exercises. After reading the theoretical part, he/she can proceed to do exercises by solving practical programming problems. If he/she wants, the student can use a text help option which is available, and typically this is an essential part of theory needed for the solution of the problem or a practical advice how to proceed for solving a concrete problem. After solving a problem, the student can ask to see the prepared solution of the problem (solution produced by a teacher and stored within ITS). After this, the student can evaluate himself (giving a mark) his work by comparing his solution and teacher's solution. Depending of his/her results, the student can choose to do more examples. When requiring to do more examples, he/she can choose to do examples designed for the same level or for the higher level (harder problems).

So, the student can choose what knowledge level to achieve (personalized targeted mark), and also to adapt the speed of learning process to his capabilities (personalized speed of learning). The student can do first the exercises for lower marks and after that the exercises for higher marks. Also, the student can skip easy examples, in the case that the student learns quickly or has some previous knowledge. All learning results gained during learning process are memorised for every student by using a data base. These results can be viewed by human teacher supervisor.

From teacher author point of view, all that is necessary is to enter lessons and exercises, but these learning materials are classified by ITS mark level and by order in learning process. Namely, every exercise is designed for a certain mark level (knowledge level), and also for a certain mark level there is several examples which order is defined

(first exercise, second exercise, third exercise) in order to satisfy the various learning speed of students. Also, for every exercise it is required from the human teacher to enter hints (text help), based on his experience with students.

So, the human teacher has the following options:

- Entering theoretical lessons
- Entering exercise problems
- Entering text hints
- Entering problem solutions
- Classification of problems
- Monitoring student results

And the student is offered the following options:

- Choosing the mark level
- Choosing the field (section) of study
- Reading theory lessons
- Solving examples
- Asking for text help (hints, suggestions)
- Viewing the teacher problem solution

- Comparing his/her solution with the teacher solution
- Giving a mark to himself/herself (evaluating his/her own problem solution)
- Memorising results of learning process

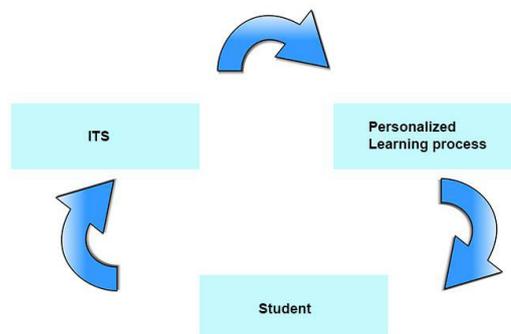


Image 5: BMU Java ITS

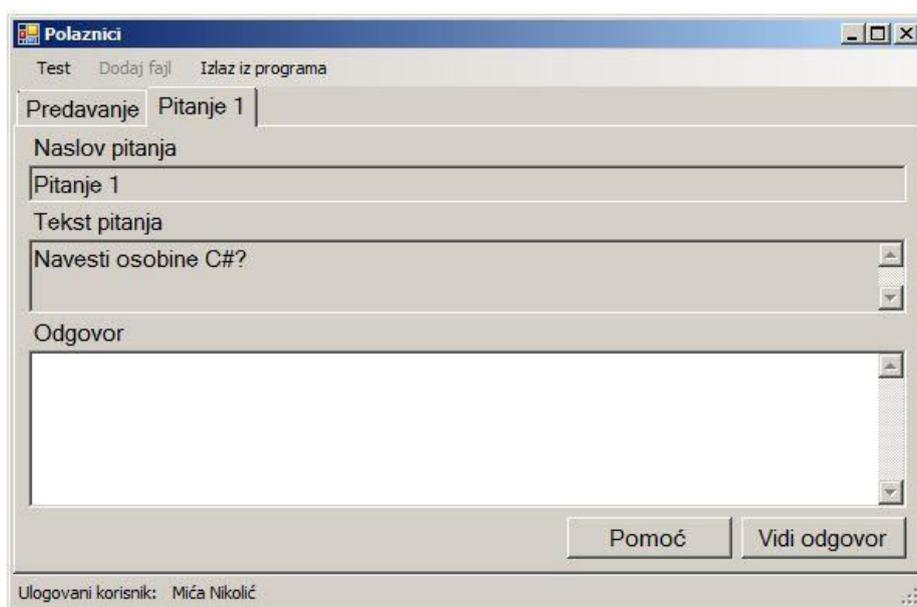


Image 6: BMU Java ITS - Start screen

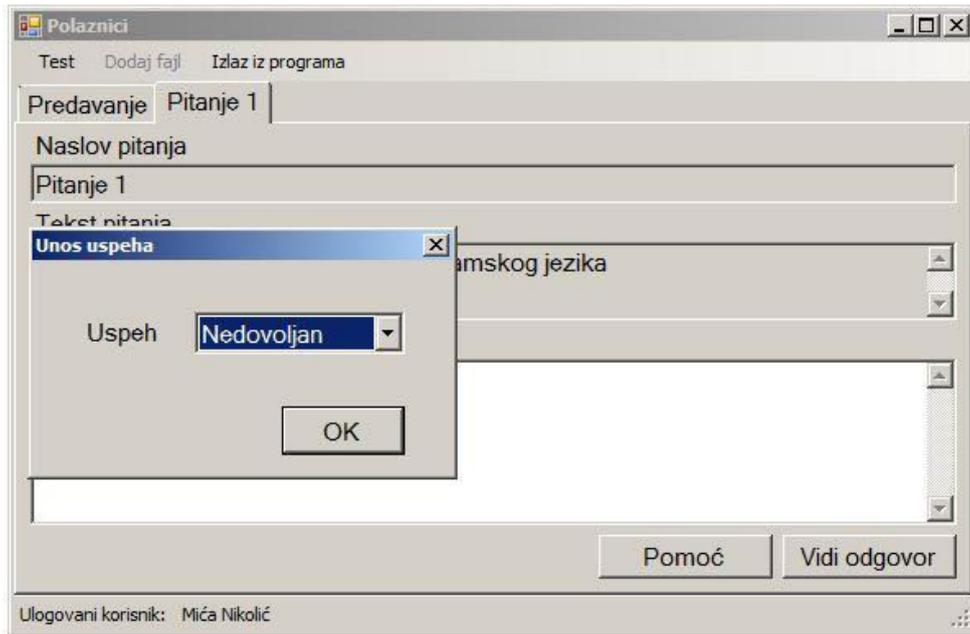


Image 7: BMU Java ITS - Grading

5. CONCLUSION

ITS is an useful tool for learning programming languages. They are typically used as an additional support to traditional learning methods. ITS systems will have certainly a more and more important role in the future, since online e-learning is spread more and more every day. Especially, personalized learning is a very important part of developing ITS, since personalized learning is more effective and exciting than nonpersonalized learning.

This paper briefly describes existing Java ITS. Also, in more details, an ITS for e-learning of Java, developed recently at the Belgrade Metropolitan University is described. This system enables personalized learning of Java. ITS further development is planned, since it is estimated that it can save significantly teacher and student efforts and time during e-learning process.

ACKNOWLEDGMENT

This work was supported by Ministry of Science and Education of Serbia (Project III44006).

LITERATURE

- [1] Intelligent Tutoring System for Learning Java Objects
<http://airccse.org/journal/ijaia/papers/0411ijaia05.pdf>
- [2] SimStudent
<http://www.cs.cmu.edu/~wcohen/postscript/simstudent-authoring-submitted.pdf>
- [3] Intelligent Tutor, Environment and Manual for Introductory Programming
<http://ijcai.org/Past%20Proceedings/IJCAI-81-VOL-2/PDF/075.pdf>
- [4] Multi-agent Intelligent Tutoring System for Learning Computer Programming
<http://www.pitt.edu/~peterb/papers/ETTI92.pdf>
- [5] Intelligent tutoring system for novice programmers
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.71.6542&rep=rep1&type=p>