CALKAS: A Computer Architecture Learning and Knowledge Assessment System

Jovan Djordjevic, Aleksandar Milenkovic, Ivan Todorovic, Darko Marinov[*]
Faculty of Electrical Engineering, University of Belgrade
P.O.Box 3554, 11120 Belgrade, Yugoslavia
E-mail: jdjordjevic@kiklop.etf.bg.ac.yu

[*] *Massachusetts Institute of Technology*
Laboratory for Computer Science
545 Technology Square #619a
Cambridge, MA 02139, U.S.A.

Abstract: *The paper presents a Computer Architecture Learning and Knowledge Assessment System named the CALKAS. It is a software tool aimed to be used for teaching Computer architecture and organization. It offers the knowledge assessment and self-learning facilities. The knowledge assessment facilities are meant to be used in laboratory for the lab test and at home for the self-test. The self-learning facilities are meant to be used at home in the process of preparation for the work in the laboratory and for the exam. The CALKAS is developed as a WWW application.*

1. Introduction

At the Faculty of Electrical Engineering, University of Belgrade, there are a few undergraduate courses in the area of computer architecture and organization. The second year course in Computer architecture and organization attended by the students of all departments covers the basic concepts such as the processor architecture and organization, the memory, the input/output subsystem and the bus [1]. The third year course in Advanced computer architecture and organization, attended by the students at the Department of Computer Science, covers topics such as the architecture and organization of CISC and RISC processors, the organization of pipelined processors, the storage system, the interconnection networks and the hierarchical memory system [2].

As a support for teaching these two courses a few educational environments have been developed and used by the students in laboratory experiments [3, 4]. This has been done with the aim to demonstrate to the students some possible practical implementations of all main theoretical concepts lectured in the classroom. Each of these educational environments contains the following elements:

- the educational system such as the educational computer system - ECS, the hierarchical memory system - HMS, etc., designed at the RTL level,
- the reference manual which includes all implementation details with appropriate circuit drawings and their explanations,
- the software package made up of the graphic simulator, which allows one to follow the functioning of the system presented in a graphical manner at the clock level, and the appropriate tools, which make it possible to initialize the system, specify test examples and run the simulation, and
- the set of laboratory experiments.

The practical work with the educational environments takes place in the laboratory where the students carry out a set of laboratory experiments. The students prepare each laboratory experiment from the reference manual by studying at home the design of a particular part of the system. Their knowledge of the topic being the subject of a particular laboratory experiment is assessed by a laboratory instructor before the access to the laboratory is being allowed. In each laboratory experiment the students follow the functioning of the appropriate educational system using the simulator and the appropriate tools. They complete each laboratory experiment by submitting written reports containing answers to questions related to some typical situations being demonstrated.

Such organization of laboratory experiments and the fact that a large number of students have been taking laboratory experiments imposed the need to develop a software tool which would automate the assessment of students' knowledge and the keeping of evidence about each laboratory experiment for each student during the course of laboratory experiments. It was also strongly felt that some kind of software tool for self-learning various topics in computer architecture and organization might be of great help to the students in the process of preparation for the laboratory experiments and the exam itself. The software tool named the CALKAS (*Computer Architecture Learning and Knowledge Assessment*

System) has been developed with the aim to meet the following two requirements: the knowledge assessment and the self-learning. The first version of this software tool was devised using Microsoft Access. However, this approach could not completely satisfy the additional requirements such as the simplicity of user interface, the security of data, the availability, the maintainability, the extendibility, etc. In order to fulfill all requirements, the new version of CALKAS is devised as a WWW application.

The rest of this paper is organized as follows. Section 2 describes the facilities offered by the CALKAS. Section 3 discusses the internals of the CALKAS. Section 4 contains a conclusion.

2. The facilities offered by the CALKAS

The CALKAS offers facilities for the ordinary users and the system administrators accessible at the user level and the administrator level, respectively. At the user level the CALKAS is used by the students, while at the administrator level by the laboratory instructors.

2.1. The user level

At the user level the students use the CALKAS in the following two modes of operation: the knowledge assessment mode and the self-learning mode.

The knowledge assessment mode

The knowledge assessment mode facilities are meant to be used in the laboratory for the lab test and at home for the self-test.

The main purpose of the Lab Test is to assess whether the students are familiar with the topic of a particular laboratory experiment to such an extent that the access to the laboratory can be allowed. In addition to that, it facilitates to keep evidence of each student's work in the laboratory during the course of laboratory experiments.

The first step that a student that wants to work with the CALKAS in the knowledge assessment lab test mode has to do is to login (Fig. 1). The student carries out the login by entering its identification number, selecting the laboratory experiment and activating the **Start** button. The system performs a check whether the student with the entered identification number is in the database of students. If so, the knowledge assessment of the appropriate laboratory experiment starts. The CALKAS generates randomly the predefined number of questions with offered answers from the database with questions and begins to count down the time remaining for giving the answers. The student answers the questions by activating the appropriate check boxes in front of the answers deemed to be correct (Fig. 2). The student can view all questions generated, give answers, modify answers already given and review the answers given by activating the **Next>** and the **<Previous** buttons. When one finds that he has completed the test, even if the time predetermined for giving the answers has not yet expired, he can submit the test by activating the **Submit** button.

The CALKAS checks the correctness of the answers given and based on the predefined knowledge threshold generates a message whether the test has been carried out successfully or unsuccessfully. If the test has not been submitted within the predetermined period of time, the CALKAS stops the knowledge assessment when the predetermined time expires and automatically submits the test. The information concerning the knowledge assessment completed, such as the student's identification number, the date, the time, the laboratory experiment identification number, the questions generated, the answers given etc. are saved in the appropriate data base tables. This makes it possible to obtain at any time during the course of laboratory experiments all relevant information concerning all knowledge assessments taken by any student from the database of students.

When a student has completed the Lab Test he can get the Lab Test Report (Fig 3.). This report contains the score and a table with all questions from the test, the answers offered for each question, the answer given by the student (marked with asterisk) and the correct answer colored in red. At any time the **Explain** option can be activated which invokes the CALKAS to give the correct answer to the current question. In the case when the question is a theoretical one, the answer is given in the form of detailed explanations. In the case when the question is related to some specific details of the educational system, a pointer to the appropriate part of the HTML version of the educational system reference manual containing the required descriptions is given.

The main purpose of the Self Test is to provide facilities, which would allow the students to prepare for the lab test and assess their own knowledge. The first step that a student has to do is to specify the topic, the number of questions he wants to answer and the time period for answering questions, and to activate the **Start** button (Fig. 4). The CALKAS uses the database with questions to randomly generate the required number of questions with offered answers and begins to count down the time remaining for giving the answers. The steps that follow and the facilities offered are similar to those available in the lab test mode. The student answers the questions and submits the test using the Self Test Exercise window similar to the one given in Fig. 2. The CALKAS checks the correctness of the answers given and generates a report specifying the number of correct and incorrect answers. It can also provide, if requested, a self-test report similar to the one given in Fig. 3.

The self-learning mode

The self-learning mode facilities are meant to be used at home. Their main purpose is to help one to prepare for the work in the laboratory. In addition to that, they facilitate one to learn various topics in computer architecture and organization during the course of preparation for the exam.

The first step that a student that wants to work with the CALKAS in the self-learning mode has to do is to specify the topic and subtopic (Fig. 5). A hierarchical structure of the course in Computer Architecture and Organization is given. At the first level a student can select one of the following topics: Processor Architecture, Processor Organization, Input/Output, Bus, and Hierarchical Memory System. At the second level the student can select one of the possible subtopics; in the example presented in the Fig. 5, Processor Architecture contains the following subtopics: Programming Registers, Data Types, Instruction Formats, Addressing Modes, Instruction Set, and Interrupts. By activating corresponding pointer students get appropriate part of the HTML version of the book intended for learning Computer Architecture and Organization. The students can view text with figures, and navigate through the book using **Next>** and **<Previous** buttons.

2.2. The administrator level

At the administrator level the laboratory instructors use the CALKAS to maintain the database of questions and the database of students. The CALKAS provides simple forms that can be used to enter new questions and update the list with answers, modify for each question the list of offered answers, add and remove the names of students, etc. (Fig. 6). In addition to that, one can define all necessary parameters for the knowledge assessment such as the number of questions, the time period within which the answers should be given, the knowledge threshold for the pass, etc. Finally, the CALKAS allows one to generate and print various reports such as the list of students that have successfully completed the laboratory experiments and are allowed to take the exam, the itemized report for each student with the status for each laboratory experiment (the knowledge assessment passed/not passed, the laboratory experiment done/not done etc.).

3. Internals of the CALKAS

The core of the CALKAS is a relational database implemented in Oracle RDBMS [5]. The database keeps information about students, questions and offered answers, test logs, and other relevant information.

For each laboratory experiment, a set of questions with offered answers is defined. For each question, a list of correct answer(s) is also defined. A support for additional graphical illustration of a question is also implemented. Besides, each question is tagged with a field that shows the degree of difficulty. This field can be determined statistically according to the students' answers during test sessions. This field is used by a tool, which automatically chooses questions for a test from a pool of questions to unify the difficulty of the test. This approach enables the laboratory instructors to set the test level desired, e.g., easy, medium or difficult.

The front-end part of the CALKAS used in the students' knowledge assessment is implemented using ASP (*Active Server Page*) and Java. ASP runs on Microsoft Web Server and uses ADO (*Active Data Objects*) for access to the database. The test starts by calling a predefined WWW page using a Web browser. In order to provide data protection, this part is implemented as a Java applet, which uses JDBC (*Java Database Connectivity*) to access the Oracle database residing on the server [6]. The Oracle's *thin-client* drivers for JDBC are used, so there is no need for any Oracle software to be installed at the client sites. The software that represents an interface to the database is implemented as a Java class. It makes it easy to move to another RDBMS system.

The self-learning part of the CALKAS also starts by calling a Web browser. This part is implemented using HTML.

The part of the CALKAS that provides the administration facilities is implemented in the same way as the one that provides the knowledge assessment facilities. These facilities are protected from the unauthorized access.

4. Conclusion

The CALKAS software package has been presented in this paper. It makes it possible:

(a) to automate the assessment of students' knowledge in lab test and self-test modes,
(b) the keeping of evidence about each laboratory experiment for each student,
(c) to carry out the self-learning of various topics in computer architecture and organization and
(d) to maintain the database of questions and the database of students.

The CALKAS has been a useful aid to the laboratory instructors in making the overall organization of laboratory experiments more efficient by significantly decreasing the need for human and other resources. The CALKAS self-test and self-learning facilities have been of great help to the students preparing for the laboratory experiments and the exam itself.

The CALKAS software package has been implemented as the WWW application. This contributed to the simplicity of an user interface and resulted in the increase of the security of data, the availability, the maintainability, the upgradeability, etc.

A possible direction of further development of the CALKAS software package are its integration both with the educational environments used in laboratory experiments in computer architecture and organization and the faculty database.

References

[1] W. Stallings, Computer Organization and Architecture, Prentice-Hall, New Jersey, USA, 1996.

[2] D. A. Patterson, J. L. Hennessy, Computer Architecture: A Quantitative Approach, Morgan Kaufman, San Francisco, California, USA, 1996.

[3] J. Djordjevic, A. Milenkovic, N. Grbanovic, M. Bojovic, “An Educational Environment for Teaching a Course in Computer Architecture and Organization,” Proceedings of the 4th Annual Workshop on Computer Architecture Education, Las Vegas, NV, January 1998.

[4] J. Djordjevic, A. Milenkovic, S. Prodanovic, “A Hierarchical Memory System Environment,” Proceedings of the 5th Annual Workshop on Computer Architecture Education, Barcelona, Spain, June, 1998.

[5] Oracle Server Administration Guide, Oracle Corporation, 1997.

[6] Oracle JDBC drivers, Oracle Corporation, 1997.

Figures

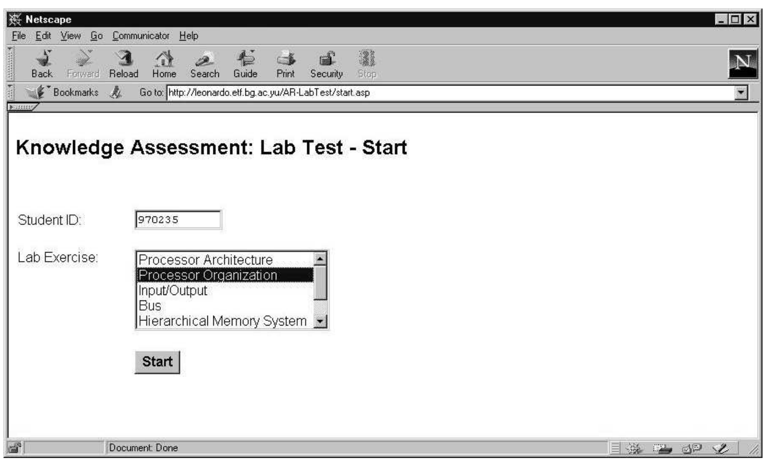

Fig. 1. Students’ knowledge assessment: Lab Test Start Window

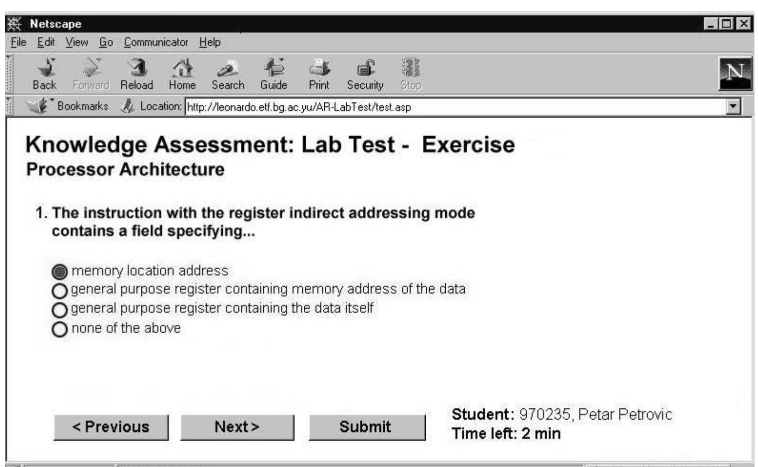

Fig. 2. Students’ knowledge assessment: Lab Test Exercise Window

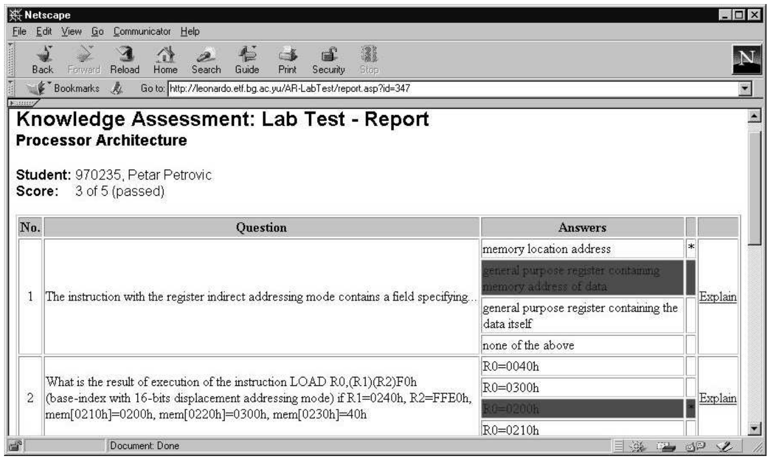

Fig. 3. Students' knowledge assessment:
Lab Test Report Window

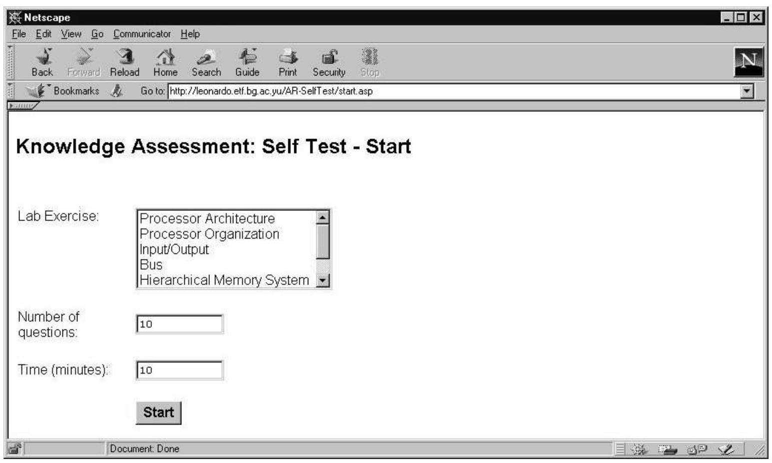

Fig. 4. Student knowledge assessment:
Self Test Start Window

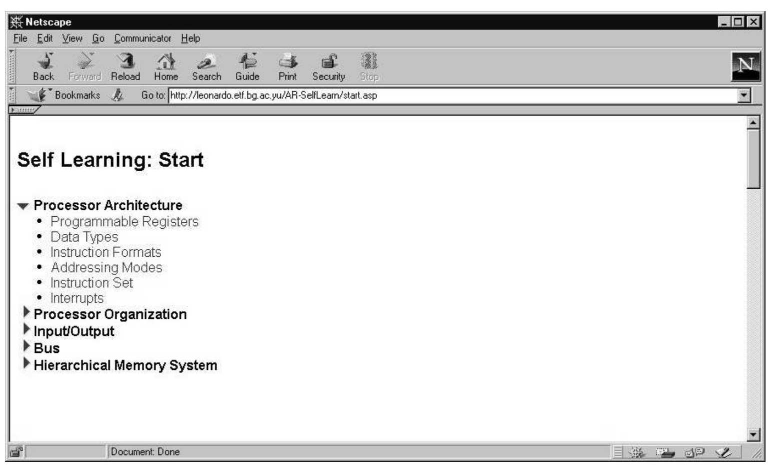

Fig. 5. Self Learning:
Start Window

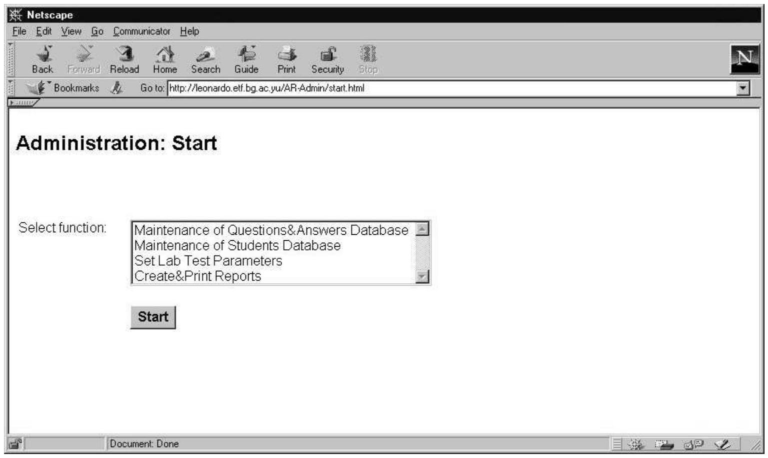

Fig. 6. Administration:
Start Window.