A New Didactic Method for Programming in C for Freshmen Students Using LEGO Mindstorms EV3

Simón Rodríguez Perez, Carolin Gold-Veerkamp, Jörg Abke, Kai Borgeest
University of Applied Sciences Aschaffenburg
Würzburger Straße 45
D-63743 Aschaffenburg
email: {simon.rodriguez-perez, carolin.gold-veerkamp, joerg.abke, kai.borgeest}@h-ab.de

Abstract—In this paper a new teaching-learning-method including the LEGO Mindstorms EV3 is presented. This computer sciences course is part of the studies of mechatronics in the first two semesters. The programming language being mediated is ANSI-C. Computer science is not the most popular subject in the interdisciplinary engineering field of mechatronics, therefore the EV3 should be included to increase motivation. For the new approach the former course – without the usage of EV3s – was evaluated and a competency profile was identified. On this basis a new didactical method was designed and implemented. The EV3 is programmed with the IDE Eclipse and with a self programmed plug-in.

Index Terms—C-programming, didactic method, EV3, freshmen students, LEGO Mindstorms.

I. INTRODUCTION

In engineering classes computer science is not the favorite subject. A lot of students have big problems with programming, because of the complexity, abstract thinking and the fact that it is not tangible. Also many of the mechatronic students prefer the field of mechanical engineering, so they like the real hardware more. This is an empirical value of several professors of the university in Aschaffenburg.

The aim is to improve the quality of teaching and learning in C-programming, taking place in the first and second semester of a bachelor course in mechatronics. The subject lists a total of eight ECTS credit points (four for each semester) and requires 180 minutes (50% lecture and 50% practical sessions) plus preparation and wrap-up time per week. Another reason is to lower the obstacles for software engineering in the fourth semester, in which the students have to program a microcontroller in a project using C.

In the following Sections an analysis of the students’ evaluation to identify potentials for improvements takes place. Additionally, a comparison of technical competencies between intention and mediation was made to detect further possibilities for optimisation.

Using this information a new haptic teaching-learning-method was developed, which is explained in this paper.

This topic was presented in earlier papers submitted to the IEEE, but the themes differ from this publications. In this course the IDE Eclipse with the programming language C is used, in contrast to [1]–[9]. Furthermore, in [1]–[5], [7], [10], [11] the LEGO Mindstorms NXT or RCX is applied, not the EV3, which offers a lot of new possibilities. Also the target group is different from the one in this paper: [1], [5], [7], [8], [12]. In [9]–[11] they have the same target group – freshmen students in engineering studies –, but an older version of the robot or another programming language.

The paper is organized as follows: In Section II there is an overview of the used software (Eclipse and our own EV3 plug-in) and hardware (EV3). Then the detected potentials for a new course are described in Section III. With this information the new teaching-learning arrangement was developed and is pictured in Section IV. To validate this new course there was an evaluation of the new exercises (Sec. V). The conclusion (Sec. VI) summarizes the content.

II. CONTEXT

A. IDE Eclipse

The IDE (Integrated Development Environment) Eclipse [13] is a powerful development environment, which is also used in industry. Also it is an open source project, which offers a lot of possibilities for programmers, e.g. developing own plug-ins. A lot of companies use Eclipse for developing software, so the students learn how to work with a practice-oriented tool. With all these points in background, the decision was made to use Eclipse for this course and in higher semesters.

B. LEGO Mindstorms EV3

LEGO Mindstorms is a product of the toy manufacturer LEGO. It contains a programmable brick, a lot of sensors and motors and can be build up with the LEGO technical line. [14]

The robot (Figure 1) has two different types of motors, one for higher speed and one with higher torque. The standard sensors are listed here:

- Touch Sensor
- Color Sensor
- Ultrasonic Sensor
- Gyro Sensor
- Infrared Sensor

978-1-4799-8706-1/15/$31.00 ©2015 IEEE 20-24 September 2015, Florence, Italy
Proceedings of 2015 International Conference on Interactive Collaborative Learning (ICL)
There are three different versions of the LEGO Mindstorms, RIS, NXT and EV3. The EV3 is the latest Version and offers a lot of new possibilities with its Linux kernel.

With the original software from LEGO the EV3 can be programmed graphically. Also there are lot of extensions for most of the common languages. Only for ANSI-C, which is used in this course, there was no complete plug-in including all the robot functions.

C. Eclipse plug-in for EV3

The reason for developing our own plug-in was that there was no possibility to program the brick with ANSI-C without changing the EV3-firmware. Also there was no complete application programming interface (API), which included all motors and sensors from LEGO.

So the existing projects were completed by including the functions to readout all sensors. It is now possible to receive all data with its correct resolution.

After completing the API a plug-in for the IDE Eclipse was programmed. It creates a new project, includes the API and the necessary cross compiler. For downloading the files via USB to the EV3 an interface was designed, which is shown in Figure 2.

III. POTENTIALS FOR IMPROVEMENT

At the end of the winter semester 2014/15 an evaluation in the first semester was performed.

The following potentials were detected:
1) More practical orientation
2) Not enough exercises
3) Missing illustrative exercises

Also the students were asked about their expectations for the second semester:
4) Get an insight of embedded programming
5) Enhancing learning-by-doing
6) Improved comprehension

With these answers in background, the decision was made to use LEGO Mindstorms Robots [14] to improve the process of learning. The EV3 is an embedded system, which offers the students a more practical and illustrative learning by moving a real hardware and selecting data from the laboratory environment.

As a consequence the students have been asked which advantages and disadvantages they would expect by the implementation of a robot into the teaching-learning arrangement.

The listed advantages have been named most often:
- Better understanding
- More practical relevance
- More fun in education
- Higher motivation
- Outcome and errors directly visible
- Incentive to work harder

Disadvantages:
- Complexity → New challenges
- Programming takes longer
- Too specific context → deflection from the main teaching objective
- Exercises can not be done at home
- Limitation of the programming topics

IV. CONCEPTUAL APPROACH

On this basis, a new teaching-learning arrangement for the practical part of the course has been designed, which tries to implement the potentials found, to meet the students’ prospects and to eliminate the expected disadvantages.

Figure 3 shows this new conceptual design by illustrating the theoretical as well as the practical section.
The theoretical session is an ex-cathedra lecture where themes like variables, functions, pointers and abstract data types are part of the course.

In the practical session the theoretical knowledge is deepened. At the beginning of the first semester the students have to program on the PC, because this approach makes it easier to focus on teaching the fundamentals and learning how to work with the environment Eclipse. The students learn how to build up a C-project, define and use simple data types (e.g. integer, boolean, float, ...) as well as declare variables and existing number systems (e.g. decimal, dual, floating point, ...).

In the second part of the first semester and at the beginning of the second semester the EV3 is used to support the learning process, which is appropriate for the following themes e.g.:

- Conditional statements: React on specific sensor values and control the flow
- Loops: Repeat specific steps, like reading out the sensors
- Pointer: Save the sensor values temporarily in an array and analyze them
- File management: Create, read and write (in) files to save the data in the long term and check them with a PC

In the second semester will end with a small project to repeat all learned topics and to use a few algorithms. The project will take approximately three weeks while working in team. One possible task could be a pathfinder for a labyrinth. For checking the implemented algorithm the design of the labyrinth will be alternated.

Subsequently there are two exercises explained exemplarily.

A. One- and two-dimensional arrays

In this exercise the students should learn how to work with arrays. Therefore they have to implement a distance meter. The robot reads out the distance to the next object with its ultrasonic sensor. The actual distance is shown on the LC display and updated cyclically. If the touch sensor is pressed, the distance has to be saved in a field, which was declarated at the beginning of the program. After finishing the measurement the robot should show the number of data points, the minimum and the maximum distance.

B. Storing data in text files

Later on, the distance meter should be extended. For this the code of the old exercise (Sec. IV-A) will be copied. Then a text file will be opened or a new one created. When this text file will be opened successfully all datapoints are saved in this file. After closing the file and finishing the program, the file can be opened at the PC and the data can be processed. This exercise should show how important it is to save data and how it is implemented.

V. EVALUATION AND EXPERIENCES

The following Section includes the perspective of the students (Sec. V-A) as well as of the laboratory team (Sec. V-B) and combines the two to give recommendations for possible modifications and adjustments. The laboratory team consists of one professor, two tutors and one masterstudent.

All Figures and explanations are based on the actual status of the semester (end of May 2015) and includes only the EV3 exercises, which were performed until now.

A. Students

After every exercise the students were asked how difficult it was using a questionnaire. The result of the difficulty on average is shown in Figure 4, thereby a five-point Likert scale (1 – really easy, 5 – very hard) was exploited. Each group consists of about 18 students.

![Figure 4. Graph of the difficulty of the exercises](image)

The graphs show a relatively equal trend. The most difficult exercises on average are structures and dynamic memory management.

The first exercise was only an introduction into the EV3 without any themes and the values show that programming the brick itself is a bit difficult to understand. It relates to the amount of commands that are necessary for programming an easy program. A solution for this problem could be a smaller list of commands, by reducing it to a minimum.

The following exercises seem to be quite difficult from the students’ point of view. But in contrast there has been no evaluation of prior semesters concerning the difficulty of exercises without the usage of EV3s.

The graphs show that in every group the fun depends on the difficulty respectively vice versa.

In Figure 5 the fun and the difficulty of the regular exercises with the EV3 are shown. The scale goes from 1 (no fun) to 5 (very much). The graphs show that in every group the fun depends on the difficulty respectively vice versa.
B. Laboratory Team

The problems the students have with programming the EV3 could be reduced to the fact that the students have big problems with using the knowledge attained in the first semester; e.g.: implementing loops, conditional statements and calling functions correctly. This can be seen during the practical session and in the evaluation.

C. Conclusion

In the first exercises the students had to program too much on the EV3, so the actual content got lost; i.e. the students were busy with understanding and implementing the EV3-specific commands. These exercises will be edited to reduce the workload of the EV3 and focus more on the topics. Also the new ones will include the EV3 only as functional instrument, so the students do not learn programming a robot but learn programming in C.

In summary, it can be said that the fun is directly related to the difficulty of the exercises. The goal for the next semester is to increase the fun factor without reducing the difficulty by improving the EV3 plug-in and shorten the workload for the EV3 programming to focus on the important content.

VI. SUMMARY AND OUTLOOK

The Eclipse plug-in was successfully implemented, including an EV3-API (Sec. II-C). This plug-in allows the students to build a program using ANSI-C Code and transfer the program onto the robot via USB-interface.

At the moment, the conceptual course design (Sec. IV) is implemented in the current semester, which started in March 2015. It is based on the identified potentials for improvement (Sec. III). In this short paper the first experiences with this new didactical method are presented (Sec. V). At the end of the semester a final evaluation of the whole course (2nd semester) will take place including all exercises. The described design for the 1st semester will be performed in the winter semester (starting in October 2015).

ACKNOWLEDGMENT

The present work as part of the EVELIN project was funded by the German Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung) under grant number 01PL12022B. The authors are responsible for the content of this publication.

REFERENCES