An Experience to use Robotics to Improve Computer Science Learning

Lady Daiana O. Maia, Vandermi J. da Silva, Ricardo E.V. de S. Rosa,
José P. Queiroz-Neto, Vicente F. de Lucena Jr.
ladydaiana@gmail.com, vandermi@ufam.edu.br, ricardoerikson@gmail.com ,
pinheiro@cefetam.edu.br, vicente@ufam.edu.br

Abstract - **In the teaching of engineering and computing, one of the major difficulties in modern education is to create an environment to motivate the student to learn, allowing them to assimilate the abstractions of Computer Science with playful and practical actions, The use of robotics in educational environments has proven to be an appropriate tool for the development of activities that involve creating, designing and planning, thereby facilitating the teaching-learning process and further expanding the integration between different areas of knowledge. Robotics in the educational context is called educational robotic and can be defined as an environment acquisition of concepts through mechanical devices, allowing development of logical reasoning and creativity, besides being a multidisciplinary tool for learning. Our goal in this work is to provide an environment for open source programming by using robotics in education and help students to improve their learning of programming languages and software engineering disciplines through a study case where we used the Lego Mindstorms Educational Kit.**

Index Terms – **Environment for open source programming, Lego Mindstorms, Robotics in the educational context.**

INTRODUCTION

Robotics is defined as the intelligent connection between the perception and the action. It is the necessary degree of intelligence to complete a particular task and involves a physical interaction between the system and the environment where the task is taking place. That is, it can react to external inputs through robots in order to execute an activity [1].

The Lego Mindstorms Kit is a set of robotics for the educational area, which lets you create and program robots, using simple mechanical parts, in order to perform simple and complex tasks. The kit consists of mounting blocks, motors, sensors, and a microprocessor is the brain of the system.

The use of robotics in educational environments has proved to be an appropriate tool for the development of activities involving creating, designing and planning, thus facilitating the teaching-learning process and extending the integration between different areas of knowledge [2].

Researches using robots in artificial intelligence teaching and robotics technology [3], and mobile robotics as a tool to support computing learning [2], have been interested in the use of robots in programming and software engineering teaching [4]. Regarding programming language learning, the robotics becomes an enabling tool for programming, giving the learner the necessary motivation to use the concepts seen in the classroom.

Free software is also an enabling tool for learning, by providing access to source code of various applications, including operating systems, sharing the knowledge. This way it is possible to use low cost tools in order to reduce the project cost, especially when it comes to educational projects. The main advantage when using this approach is the possibility of a customized code source, and the application of reuse techniques to take advantage of the only interesting parts of the project [5].

This paper presents a study case on the Software Engineering discipline taught to graduate students at the Universidade Federal do Amazonas. Robotics is used as a tool for the development of the final project of that discipline. In this project, students will have to develop an embedded application to navigate a maze, using the Lego Mindstorms Kit to mount a robot, and basics of programming to implement algorithms. The project aims to improve the students learning level in Software Engineering and Programming Languages.

Section 2 presents the robotics in educational context. Section 3 presents the methodology used in the discipline in both theoretical and practical lessons in laboratory. Section 4 shows robotics as programming environment and software tools used to build the environment. Section 5 describes the details of the discipline's final project used in study case and a description of the steps taken during the implementation. The results are shown in Section 6. Finally, Section 7 presents the final comments.

ROBOTICS IN EDUCATIONAL CONTEXT

Robotics in educational context is known as pedagogical robotics and can be defined as an adequate environment for one to acquire concepts through mechanical devices, allowing him to develop the logical reasoning and creativity, being also a multidisciplinary tool for learning [6].

Experiences with educational robotics are getting popular in Brazil. There are some initiatives of pedagogical projects using robotics, such as the Núcleo de Informática Aplicada à Educação (NIED/UNICAMP) developing a Pedagogical Robotics project in the area in Education. This project aims to expand and diversify the use of computer as a support tool in learning and teaching processes and one of its goals is to spread the use of computers in state schools, municipal ones, and universities in Brazil [7].

Another initiative in this respect is the Free Robotics Project created by CEFET-MG that does not use standardized kits and search for free solutions in replacement for commercial products. The project proposes the use of free technologies as the basis for programming and uses electronic scrap equipments for robots construction [8].

According to [9], the main kits supplier in Brazil is Lego Educational. The Lego's methodology is based on Jean Piaget's constructionism. The pedagogical proposal was developed by Seymour Papert of the Massachusetts Institute of Technology (MIT) and allows the students the subjective construction of knowledge through technology.

When we reflect on the pedagogical robotics we are thinking about programming. When programming a robot to perform a task, it is necessary that each action of the robot be connected to a function in a logical and ordered way. The logic is inescapable to the apprentice that needs to assemble, program and test his robot. For that we need to know the symbolism for this programming.

METHODOLOGY

The methodology used in the discipline is based on [10]. During the experience development we divided the class into three teams of six or seven components to develop the academic work, putting into practice the theory gained in classroom. In the theoretical lessons we used whiteboard, brush and data show projector. In practical lessons, we used two Lego Mindstorms Educational kits to build simple prototypes and to use algorithms implemented in Java and C++ programming languages.

The practical lessons were accompanied by a monitor, responsible for coordinating the laboratory teams, and consisted of tasks performed by students of the discipline, focusing on the installation of the programming tools, IDE (Integrated Development Environment), plug-ins, operating systems, and other tasks related to Lego programming environment.

During the practical lessons, simple algorithms were constructed to test the logics for the use of both IF and ELSE conditional commands: FOR, WHILE, REPEAT repetition commands and other special features of programming languages. Thus, it was possible to test the performance of the proposed algorithms, which allowed the execution, testing and code refactoring directly into the device, in order to improve the algorithm.

After this stage, the final work of the discipline had been defined. The teams received a proposal for working on software to program the Lego Mindstorms in order to travel a maze using light sensors, as Figure 1. The maze contains an entry and an exit, identified through gray scale. It also has junctions in different gray scales in order to complicate the robot's journey.

The main goal was to organize the team's components in development teams, giving them classic roles of software engineering that is, defining the roles of project management, configuration management, analysts, developers and testers, following the Waterfall Methodology, culminating in a dispute between the teams to motivate students. The winner was the team which had the faster algorithm to first find the maze exit.

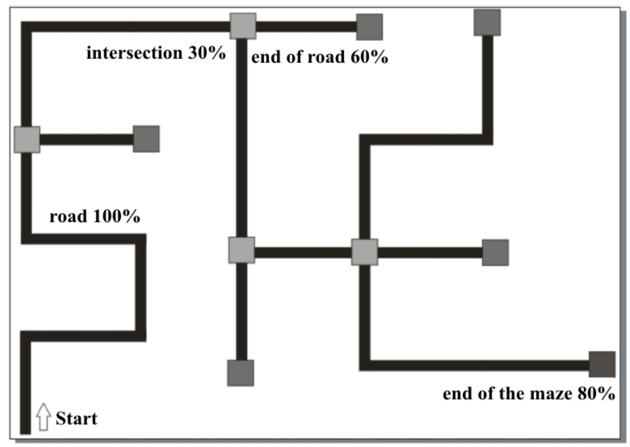

FIGURE 1
MAZE WITH A DESCRIPTION OF THE PERCENTAGE OF GRAY SCALE

ROBOTICS AS PROGRAMMING ENVIRONMENT

A robotics environment for education consists in the interaction among teachers, students and tools that provide assembling, programming and controlling of mechanical devices, characterizing a pedagogical robotic environment with the purpose of acquiring knowledge by using this picture. This programming environment was built in the robotics and industrial automation lab at the Universidade Federal do Amazonas and was chosen to use free tools, because of its advantages in a cost/ benefit comparison.

The use of free tools in the project was motivated by the fact that the Free Software is more and more consolidated and stronger, allowing the construction of a software by using parts of another, in order use and optimize the use of the code, with no need of building all from the start [5]. The advantage of using the Free Software is the possibility that you have to modify it and distribute it by sharing the acquired knowledge, besides approaching students and tools that are used everyday in software development companies, bringing students closer to a real development environment. In this topic, we will present the free technologies used in the programming environment and will give a brief description of each one of them.

I. Linux UBUNTU Operating System

The word Ubuntu is from Africa and means “humanity to others” [11]. The Ubuntu operating system, developed by Linux, uses the Linux nucleus, created by Linus Torvalds, and today it has a grand ramification in the Operating Systems world.

The Ubuntu is licensed by General License Public – GLP. With this license, anyone can use, study, modify and distribute it according to the license terms. The system was chosen for the following characteristics: robustness and trustworthiness, for having an open code, compatibility, security, speed and stability [12].

II. BrickOS Operating System

The BrickOS, licensed by GPL, is an open alternative source for the Lego Mindstorms operating systems kit, with the possibility of the source code adaptation and adequacy. It was developed from the Portable Operating System Interface pattern [13], as a solution to the limitations of the software kit called Not Quit C – NQC. The NQC is multitask and allows a larger control of the CPU of the RCX [2]. The BrickOS is distinguished due its advantages, according to the Table 1.

TABLE I
EVALUATION OF THE SYSTEMS

Systems	BrickOS	NQC
Hardware Control	Complete	Via Library
Language	C/C++	C
Execution Speed	Maximum	Medium
RCX Operating System	Proper	Lego
PC Operating System	Linux	Windows/Linux

III. IDE Eclipse with C/C++ Development Tooling – CDT

Eclipse is one IDE very used these days, because it allows the integration of many tools in only one development environment. It is a very flexible tool, and it was developed in Java by the Eclipse community. It uses Standard Widget Toolkit – SWT to display GUI elements, the SWT implementation accesses the native GUI libraries of the operating system using Java Native Interface - JNI in a manner that is similar to those programs written using operating system-specific APIs.

Although the Eclipse is mostly a Java development environment, its architecture guarantees support to other programming languages such as Java, C, C++, Phyton, PHP, Pearl and Cobol [14]. This platform was chosen due to some characteristics, such as:

- It’s a Free Software;
- It helps the development and construction of applications;
- It integrates purification instantaneously, helping to find compilation defaults;
- It has friendly environment;
- It has a portable and extensible platform, allowing the addition of other tools.

The CDT is a open source project licensed by GPL, executed exclusively in Java programming language as a set of plug-ins for the Eclipse Platform SDK. With these plug-ins, it is possible to support the application development in language C/C++ along with the advanced edition and support to the purification process [15].

PROJECT OF THE MAZE

We divided the project in two phases. First we chose the mechanical part of the robot and then we made an evaluation test to select the best prototype. In this phase we used the Lego RCX kit 2.0.

In phase two we had to build the software to robot control following the accepted methodology, using a requirement document. Then, we divided the teams in an academic project, using programming language C, C++ and Java to build the software.

After the requirement analysis, the development environment was configured and we divided the tasks among the team applying the Waterfall model presented in section Methodology.

The prototypes chosen were the ones which looked like war tanks. However, due to the amount of parts and drivability of the robot, the prototype was discarded. Through bibliographic searches, a prototype was identified as a tricycle, which was later adapted as a tractor plow, which behaved very well concerning stability and control, being one of the prototypes used in the project. Figure 2 and Figure 3 show two robots built by the teams, which were used in the experiment.

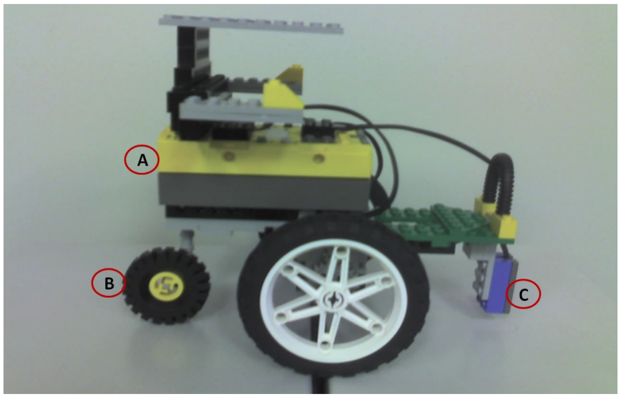

FIGURE 2
ROBOT TRACTOR, MOUNTED WITH LEGO MINDSTORMS KIT USING THE BRAIN OF THE ROBOT (A), REAR WHEEL FOR STEERABILITY (B) AND THE BRIGHTNESS SENSOR (C).

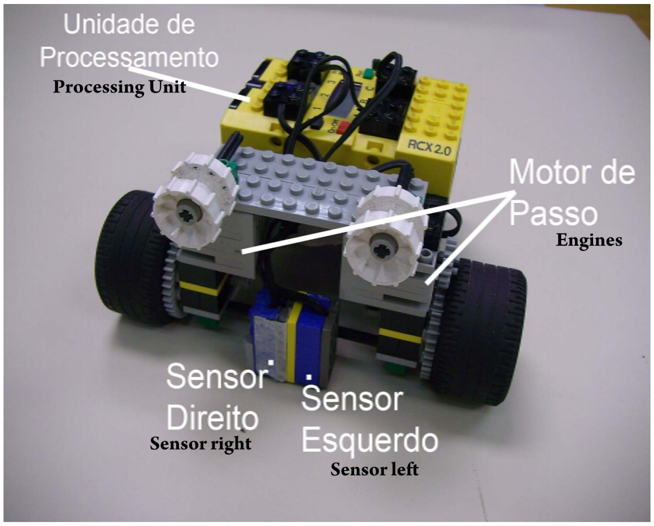

FIGURE 3
ROBOT USING SENSOR LEFT AND RIGHT.

The implementation was modularized from the construction of artifacts and integration of features, ending with a prototype robot and the software loaded to control its operation. The Table 2 gives a summary of the steps used in the project.

TABLE II
STEPS TO DEVELOP THE FINAL PROJECT

Phases of the Project	**Tasks**
1	Installation and configuration of the environment
2	Analysis of Requirements
3	Prototype using RCX 2.0
4	Requirements Implementation
5	Tests
6	Source code Refactory
7	Re-test
8	Software Delivery (race of robots)

The classes and functions implemented, facilitated the training in selected languages (C, C++ or Java), since it requires knowledge about the characteristics of language such as bonds, conditional structures, declaration of variables, among others, allowing the direct application of logic in the programming language chosen. At the end of the term, the work was presented in the form of seminars where the teams described the results, culminating with a competition between the robots built.

The objective of the competition was to identify the best algorithm and the best robot that fulfilled the task in less time through the maze to its exit. Through reports completed by students, it was possible to identify the difficulties encountered in this type of software development and the improvement of teaching-learning process in relation to programming. These improvements will be presented in the results of this work.

RESULTS

The Figure 4 shows a team of students during the competition. From the total of teams, 19 teams completed successfully the task and 5 completed partially the task. The winning team completed the task in 10 seconds with the lowest way to be traveled.

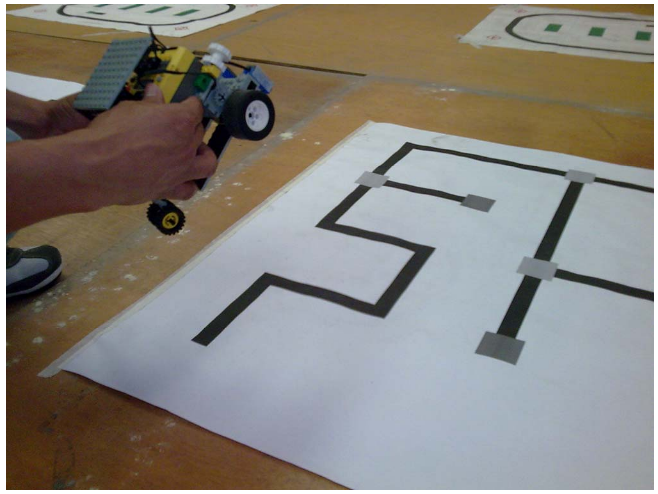

FIGURE 4
A TEAM OF STUDENTS APPLYING THEIR SOLUTION IN THE COMPETITION AND REALIZING THE TASK THROUGH THE MAZE

The experiment was conducted with 120 students divided into four periods. Seven students failed in the end, whether by withdrawal or disapproval, the other ones obtained a rate of approximately 94% of approvals. There is empirical evaluation of some advantages of using a robot to support learning, including the development-based projects, which allowed the students the ability to plan, work in teams that provided the exchange of knowledge and integration of collaborative learning programming applying the concepts seen in the classroom.

Moreover, we could perceive that the task enhanced students' motivation to see the robot working according to commands and functions planned for them. This enabled a suitable learning environment for students to develop an algorithm for the robot through the maze, by error checking the program, as the robot trajectory, allowing the code modification to fix that path. These procedures solidified the concepts of logic programming acquired in classroom.

Compared with traditional methods of teaching-learning programming, the use of the action game in robotics benefits the students who develop self-learning in order to overcome the challenge that was placed within the educational context, by incorporating the dialectic theory and practice.

In the final work, the results obtained were satisfactory, since all teams have reached the goals of discipline, through cooperation and exchange of knowledge, leveling it in the field of content related to programming language and software engineering.

The application of the concepts of object orientation and the use of programming languages (Java and C++), allowed the team this paradigm in order to use the knowledge acquired in other disciplines that relate directly to computing.

FINAL CONSIDERATIONS

This work presented a programming environment of Open Source with the Lego Mindstorms kit, using available technologies, techniques of Software Engineering and concepts of logic programming. The main advantage of the use of robotics in learning is the motivation it provides to the student and the benefits already mentioned in the result. However, there must be a detailed study aimed at applying the methodology in disciplines related to Computer Engineering, Computer Science and Electrical Engineering courses.

The case study demonstrates the use of these technologies and validates the importance of robotics as a method of learning. Therefore, it is perceived that the study of robotics can be used in various areas of education, where it assists the teaching-learning process of several disciplines such as programming, software engineering or others, providing greater motivation and, above all, the solidification of the concepts.

ACKNOWLEDGMENT

We would like to thank very much the UFAM for the use of the laboratorial infrastructure, to the CNPq and FINEP for the financial support, and to Ana Célia Said for your help in revise the text. We would also like to congratulate the students who dedicated to the tests of the developed experience.

REFERENCES

[1] PIO, J.L., CASTRO, T. H., JÚNIOR, A. N. C. (2006) A Robótica Móvel como Instrumento de Apoio à Aprendizagem Computacional. XVII Simpósio Brasileiro de Informática na Educação. Page(s): 197-206.

[2] BAGNALL, B. (2007) Maximum Lego NXT. Building Robots with JavaTM Brains. Ed. Press Variant.

[3] NOURBAKHSH, I.(2000) Robots and Education in the classroom and in the museum: On the study of robots, and robots for study. Proceedings of the Workshop for Personal Robotics for Education, IEEE International Conference on Robots and Automation, IEEE Press.

[4] JAZDI, Nasser ; LUCENA Jr, V. F. ; GÖHNER, P. (2006) Unibral, an Education and Research Cooperation between Brazil and Germany. In: 36th Frontiers In Education Conference, 2006, San Diego. Proceedings of the 36th Frontiers In Education Conference, 2006. p. 20-25.

[5] ALMEIDA, E. S., Alvaro, A., GARCIA, V. C., MASCENA, J. C. C. P., ARRUDA, B. V. A., NASCIMENTO, L. M., Lucrédio, D., and Meira, S. L. (2007). CRUISE – Componet Reuse in Software Engineering. http://cruise.cesar.org.br/, Recife, Brazil.

[6] CASTILHO, M. I. (2007) A Robótica e o Raciocínio Lógico. Available in <http://www.pucrs.br/eventos/desafio/mariaines.php#raclog> Accessed in August 2008

[7] NIED (2008), Núcleo de Informática Aplicada a Educação. Available in <http://www.nied.unicamp.br>. Accessed in August 2008.

[8] ROBOTICA LIVRE (2008), Robótica Livre. Available in <http://www.roboticalivre.org/portal/>. Accessed in September 2008.

[9] REDE, A Rede. Available in <http://www.arede.inf.br/index.php?option=com_content&task=view&id=1344&Itemid=99>. Accessed in August 2008.

[10] LUCENA, Jr, V. F. ; BRITO, Alysson ; GÖHNER, P. ; JAZDI, Nasser . (2006) A Germany-Brazil Experience Report on Teaching Software Engineering for electrical Engineering Undergraduate Students. In: 19th Conference on Software Engineering Educationand Training (CSEE&T), 2006, Honolulu. Software Engineering Education and Training CSEET 2006, v. 1. p. 69-76.

[11] UBUNTU (2008), Learn moure about Ubuntu. Available in < http://www.ubuntu.com/products/whatisubuntu> Accessed in August 2008.

[12] UBUNTU-BR (2008), Comunidade Ubuntu. Available in < http://pt.wikipedia.org/wiki/Ubuntu_Linux> Accessed in July 2008.

[13] POSIX (2008), Portable Operating System Interface. Available in <http://pt.wikipedia.org/wiki/POSIX>. Accessed in July 2008.

[14] LOSANO, F.(2008) Eclipse no Mundo Open Source. Available in <http://web.teccomm.les.inf.pucrio.br:8080/eclipse/files/EclipseDay2/palestras/eclipseoss.pdf> Accessed in July 2008

[15] ECLIPSE (2008), Comunidade Eclipse. Available in <www.eclipse.org> . Accessed in July 2008.

[16] CASTRO, Thais H. C.; FUKS, H., CASTRO Jr, A. N.; SPÓSIT, M. (2007) A. Integração de Ferramentas para Acompanhamento da Aprendizagem de Programação. Departamento de Ciência da Computação da Universidade Federal do Amazonas – DCC/ UFAM. Workshop de Ambientes de apoio à Aprendizagem de Algoritmos e Programação, SBIE 2007.

AUTHOR INFORMATION

Lady Daiana O. Pinto, Master student in computer science, computer science department - Universidade Federal do Amazonas, ladydaiana@gmail.com.br

Vandermi J. da Silva, Master student in computer science, computer science department - Universidade Federal do Amazonas, vandermi@ufam.edu.br

Ricardo E. V. de S. Rosa, Master student in electrical engineering, electrical engineering department - Universidade Federal do Amazonas, ricardoerikson@ufam.edu.br

José Pinheiro de Queiroz-Neto, D.Sc., Professor at Centro Federal de Educação Tecnológica do Amazonas and Universidade Federal do Amazonas – UFAM, pinheiro@cefetam.edu.br

Vicente F. Lucena Jr., Dr.-Ing, Professor at Centro Federal de Educação Tecnológica do Amazonas – CEFET-AM, and Universidade Federal do Amazonas – UFAM, both in Manaus. He has been developing research activities at CETELI – UFAM, lucena@cefetam.edu.br or vicente@ufam.edu.br