Teaching Programming with Lego RCX Robots

Ka-Wing Wong
Eastern Kentucky University
Richmond, KY 40475

Abstract

A common practice in most programming curricular is for students to learn computer programming in an Integrated Development Environment (IDE). Even though IDEs provide good program development support, very often, what students learn is limited to what an IDE provides. The limitation can be overcome by using additional learning activities in class. LEGO RCX robots can be used to provide additional learning activities that IDEs do not provide. We used the LEGO RCX robots in our programming classes at three different programming skill levels (introductory, intermediate, and advanced) and found that they are very useful in enhancing our curriculum.

Keywords: Lego, RCX, robot, programming, curriculum.

1. INTRODUCTION

Computer programming is a discipline that students learn problem solving through the interaction of a computer. Currently, such interaction is taken place in Integrated Development Environments (IDE). IDEs have good programming development support and many students depend on them as the main learning environment. Because of a high dependence on IDEs, students do not see many problem solving aspects of computer programming outside the IDEs. Recent effort of using robotic projects to teach artificial intelligence and robotic technology (Meeden 1996; Kumar 1998; Beer 1999; Nourbakhsh 2000) has generated interest in incorporating robotic projects in teaching programming and software engineering. Giolma and Wolz used robots to teach project design (Giolma 2000; Wolz 2001). Becker and Bergin *et. al.* use simulated robotic software in their programming courses (Bergin 1997; Becker 2001) . We investigated the use of robotic projects in class and found that LEGO RCX robots are good learning tools for students to learn computer programming. We used the LEGO RCX robots in our programming classes at three different programming skill levels (introductory, intermediate, and advanced) and found that they are very useful in enhancing our curriculum.

2. LEGO RCX ROBOTS

The LEGO RCX robot contains an 8-bit computer housed in a LEGO programmable brick to which input and output devices can be attached. Input devices include touch sensors, light sensors, and angle sensors. Output devices include motors that can turn at variable speeds and reverse direction. Traditional LEGO pieces can be attached to the motor to perform many forms of robotic movements. Programs are loaded via an infrared device called the IR tower which is connected to a personal computer. Program development can be done in several languages including NQC (Baum 2000), LegOS, and Spirit.OCX with Visual Basic.

3. TEACHING PROGRAMMING WITH RCX

We selected classes at three different programming skill levels (introductory, intermediate, and advanced) to experiment with using RCX robots to teach programming. For each class selected, we dedicated three weeks in the semester to apply a hands-on closed lab teaching approach doing RCX robotic projects with the students. At the end of the three weeks period, we gave students a test to see how much they learned from doing the robotic projects. We compared their test scores with the test scores of a similar test in another class to measure the results of our quasi-experiments.

Introductory programming level

At the introductory programming level, we selected a C++ programming class to do the robotic projects. The class has 17 students. They worked in groups of

two to three students to learned objects, control structures, and modular programming. They used the RCX robots to learn the concepts and selection of objects and control structures (decisions and loops) to solve specified problems. The problems they solved include a bump-and-turn problem in which a robot hitting an object will move back a short distance and make a turn to move forward in a new direction. The onboard program that the students wrote contains many information gathering and control decision processes. Students experimented with various objects to store information and various control structures to control the movement of a robot.

Intermediate programming level
At the intermediate level, we selected a data structure class to do the robotic projects. The class has 11 students. They worked in groups of two to three students to design and implement a data structure that can be stored in the RCX robot. The data structure is to record information necessary for directing the robot to complete a task. One of the tasks we asked the students to do is to make the robots to solve an object avoidance problem. The robot starts moving from one side of a room. Whenever it encounters an obstacle, it records the location of the obstacle in its data structure stored in the RCX. The robot then moves around the obstacle and continues on, recording the location of all obstacles it found. After the robot reached the other side of the room, we put it back to the starting point and let the robot moves again. The robot follows the recorded path traveling to the other side of the room without hitting the obstacles on its path. A nontrivial challenge here is the timing of the robot movement so that the robot does not hit the obstacles while it moves on the recorded path.

Advanced programming level
At the advanced level, we selected a senior/graduate level class in artificial intelligence. Students were asked to design a computation model that can be stored in the RCX. The challenge here is to make use of the limited amount of onboard memory in the RCX to implement the computational model. One group of students designed and implemented a neural network on the RCX robot. The robot was trained to perform a task afterward.

4. RESULTS

Students showed very high interest and enthusiasm in learning through programming the LEGO RCX robots. Many students worked in the lab after class time. The problem solving elements that they learned include:

Real timing control.
Students learned that while they are interacting with a computer through the keyboard and monitor, they have all the time they need to enter a response. There is no time limit. They can even look up reference or discuss with other students before continuing with the program. Many problems, however, do not allow such slow feedback. An onboard program that controls the movement of a robot, for example, requires immediate feedback.

Resources management.
Within an IDE, students do not realize that problem solving require good use of available resources. Many students sit in front of the monitor and type up a program without thinking whether there is a better solution. Problem solving in the robotic projects makes them realize good programming require planning and good utilization of resources.

Awareness of problem environment
Traditionally, students do not have to consider the program running environment when they design their program. IDEs take care of all of the problems. However, with a robot, a program may not work correctly when there is an environmental parameter change. Examples are changes in the size of the wheel and gear to motor ratio. Many students experienced some mechanical problems when testing their programs. One group of students wrote a program to make a robot to move backward when it hits an object. Their program worked when the robot bumped into something soft (like the hand of a student). However, when the robot hit a wall, it broke into several pieces.

Development of modular solution
Many students found out that robots are systems that can have modular units. There usually is a power module, a gearbox module, a wheels and axle module, etc. The modular structure of the robot helps students visualize the modular structure of a program and modular approach in problem solving.

Student seemed to retain the learned knowledge longer and they did better in test questions related to the materials they learned through programming the LEGO RCX robots. We gave a test to the students after the three weeks experimental period and found that at the introductory programming level, the class average test scores is 2% higher than the class average score of a similar test in another class that did not use the robotic projects. The class average test score is 4% higher at the intermediate level.

We asked the students to fill out a questionnaire at the end of the three weeks experimental period. The questionnaire includes questions on the subject matters they learned, the effectiveness of learning subject matters, the effectiveness of learning teamwork, their overall opinion of learning through the LEGO RCX

robots, and their preference about learning through the LEGO RCX robots versus other interaction. We analyzed the survey results and found that students responded with very positively comment on all the questions. In particular, they all prefer learning programming through the LEGO RCX robots.

5. FUTURE WORK

We are currently studying the integration of the RCX robots with other devices to provide students with a broader learning environment. The devices include multimedia processing devices, wireless control devices, and communication devices.

6. CONCLUSIONS

Robotic project is a powerful learning tool for learning computer programming. We introduced robotic projects at the introductory, intermediate, and advanced programming level classes and found that they greatly enhanced students' learning at all three levels. Students learned real time computing, resources management, awareness of problem environment, and development of modular solution through LEGO RCX robots.

7. REFERENCES

Baum, D., 2000, Dave Baum's Definitive Guide to Lego Mindstorms. Apress.

Becker, B.W., 2001, "Teaching CS1 with Karel the Robot in Java." Proceedings of the 32nd SIGCSE, ACM Press, pp. 50-54.

Beer, R., H. Chiel, and R. Drushel, 1999, "Using autonomous robotics to teach science and engineering." Communication of the ACM Volume 42, Number 6, pp. 85-92.

Bergin, J., S. Mark, R. Jim, and P. Richard, 1997, Karel++: A Gentle Introduction to the Art of Object-Oriented Programming. John Wiley & Sons.

Giolma, P., F. Aminian, and D. Ibaroudene, 2000, "An autonomous robot – the ideal design project." Computers in Education Journal, Volume 10, Number 1, pp. 62-67.

Kumar, D. and L. Meeden, 1998, "A robot laboratory for teaching artificial intelligence." Proceedings of the 29th SIGCSE, ACM Press, pp. 341-344.

Meeden, L., 1996, "Using robots as introduction to computer science." Proceedings of the ninth Florida Artificial Intelligence Research Symposium, pp. 473-477.

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.

Wolz, U., 2001, "Teaching Design and Project Management with Lego RCX Robots." Proceedings of the 32nd SIGCSE, ACM Press, pp. 95-99.