PRACTICAL PROGRAMMING FOR ALGORITHMS AND PROGRAMMING COURSE

Conference Paper · March 2018

DOI: 10.21125/inted.2018.0786

4 authors, including:

Ayse Aybike Seker
Gebze Technical University

Onur Öztürk
Bülent Ecevit Üniversitesi

PRACTICAL PROGRAMMING FOR ALGORITHMS AND PROGRAMMING COURSE

A. A. Şeker[1], O. Öztürk[1], A. Köksoy[1], T. Gözel[1]

[1]Gebze Technical University (TURKEY)

Abstract

The importance of hardware and software combination in electronics engineering is rapidly increasing. The expected programming skills from an electronics engineer is on high level accordingly. In electronics engineering education, it is a necessity to make innovations in teaching of programming languages. To provide expected skills, the Algorithms and Programming course given at the beginning of electronics engineering education is required to be revised. Developing an instructional module that facilitate the understanding of the programming for laboratory sessions of the course may satisfy this need. In this study, the instructional module is created on an experimental platform with the microprocessor-based card, Arduino. With this platform, visual and physical experiments are brought into use of students in the laboratory. The physical counterparts of the programming knowledge are provided to the students to learn easier and have an insight for programming.

In electronics engineering education, increasing the number of students who have skills in programming from early stage will also provide success in further studies.

Keywords: Arduino, introduction to programming, C/C++, instructional module, electronic engineering.

1 INTRODUCTION

Teaching programming is to make students gain the ability of developing algorithmic solutions to engineering problems as well as adapting them to software using by electronics engineers. With rapidly increasing importance of hardware and software combination in electronics, classical teaching of algorithms and programming course is needed to be enhanced. An instructional module that facilitate the understanding of the programming for students may give them the insight how to control physical systems that they are supposed to manage in real life engineering problems.

Students face difficulties to start with a blank page and create an algorithm to solve a problem with using the programming skills they gain during programming course. New teaching methodologies might help the student to overcome their initial difficulties. Several studies have proposed the use of physical modules in teaching programming. Various applications and studies have been carried out in this context [1,2]. It will be possible to come up with the challenge of understanding programming with new teaching techniques. With the use of the physical computing paradigm, computations go out of the screen to interact with students in the real world [3,4].

Many studies have been carried out on the feasibility of physically sampling in computer programming teaching [5]. However, since most of them are oriented to programming a robot, they may not always be suitable for students without design ability at their first year in higher education. As an alternative approach, small and simple modules can be developed using electronic cards. In [6] several learning modules are designed and implement using physical computing paradigm as it shows this approach is less costly as well as easier to understand, reproducible and reliable.

In Algorithms and Programming course C/C++ is taught with classical methods. The purpose of this study is to design instructional modules with Arduino platform that will be conductive to make physical sampling of classical programming course content for students. These instructional modules are used to support laboratory sessions of Algorithms and Programming course. In second part of this paper three instructional modules are given as example to make the methodology clear. And an instructional module experimental card is designed to shorten the time students spend with setting the physical module before they start coding. As a result, the designed experimental card is exemplified for further physical computing possibilities.

2 METHODOLOGY

In Algorithms and Programming course for electronic engineering students, the concept of physical paradigm is decided to be applied. First, traditional teaching method is used in lectures and laboratory sessions. Then several instructional modules are designed to be used in laboratory sessions to see the difference in interest of students into programming.

These modules are used in C / C ++ programming language teaching. Modules are designed with a microprocessor board, Arduino, which is easy-to-use and well-equipped. Arduino is an open platform with a huge user community that is often used in the field of education [7]. Also, it offers a variety of projects, ideas and solutions in a wide range from simple circuits to complex robotic work, as well as easy use.

With the modules created using Arduino platform, students will be able to make visual samples in the laboratory. The programming process taught in the course will be carried out on various examples. Programming functions such as condition tests, loop structures, and arrays will be tested on modules built with physical elements such as LEDs, sensors, displays. For this reason, various modules are designed according to the course content. While instructional modules are designed, various physical elements that are appropriate to the course content and simple enough to use for students with limited programming and electronic component knowledge are chosen.

To apply the instructional modules in laboratory two leaflets are prepared; one for student one for lecturer. All the codes, general information about components and answers of possible student questions and problems in the laboratory session are included in lecturer leaflets. Brief information about Arduino, basic circuit theory rules and electronic components is given to students. And each experiment leaflet has the information of components used in that module.

2.1 Laboratory session with instructional modules

Laboratory sessions are explained here. First experiment is prepared to use if, nested if/else and switch case statements. To make students familiar with components a simple one is chosen. A RGB LED, and a resistor and potentiometer are used. For the next 2 experiment LCD is chosen to use the statements for, while, array and pointers.

2.1.1 Experiment 1: if, nested if/else, switch case statements for RGB LED

Circuit components are Arduino board, RGB LED, potentiometer, breadboard and 2 resistors. The components, what they are and how they work are explained in detail to students. And role of the electronic components are tried to match with traditional laboratory session tasks. For example, potentiometer, here is the value they asked from user to decide which condition to run. Instead of asking user to enter a number between a range, they do it physically by turning the knob of potentiometer. Here is the explanation of components for Experiment 1:

- *RGB LED looks like a regular LED, it has three LEDs inside the usual LED package, one red, one green and one blue. By controlling the brightness of each of the individual LEDs you can mix the colors.*
- *A potentiometer is a simple knob that provides a variable resistance, which we can read into the Arduino board as an analog value. By turning the shaft of the potentiometer, we change the amount of resistance on either side of the wiper which is connected to the center pin of the potentiometer. When the shaft is turned all the way in one direction, the voltage at its pin is zero volt, and we read 0. When the shaft is turned all the way in the other direction, the voltage at its pin is five volts and we read 1023. analogRead() returns a number between 0 and 1023 that is proportional to the amount of voltage being applied to the pin.*

In Figure 1 circuit schematic is shown. Students are asked to set up their circuit as on the schematic. This process takes time and students might make wrong connections that they realize when their code doesn’t work. To avoid this circuit doing this process step by step with student in laboratory session would be advantageous.

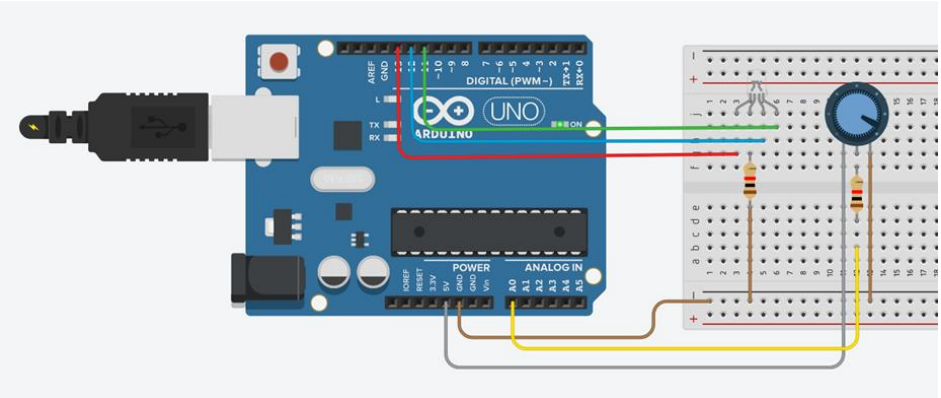

Figure 1. Circuit Diagram of Experiment 1.

The code written with using if statement is given to students that make the RGB LED blinks in the color they want by changing the value of potentiometer. And it is asked from them to rewrite the code using nested if/else statement and rewrite the code using switch case statement and run.

2.1.2 Experiment 2: for, while, array statements for LCD

Circuit components are Arduino board, LCD, breadboard, resistor, potentiometer.

- *LCD is used to display result of the code.*
- *Potentiometer is used to control contrast of LCD.*

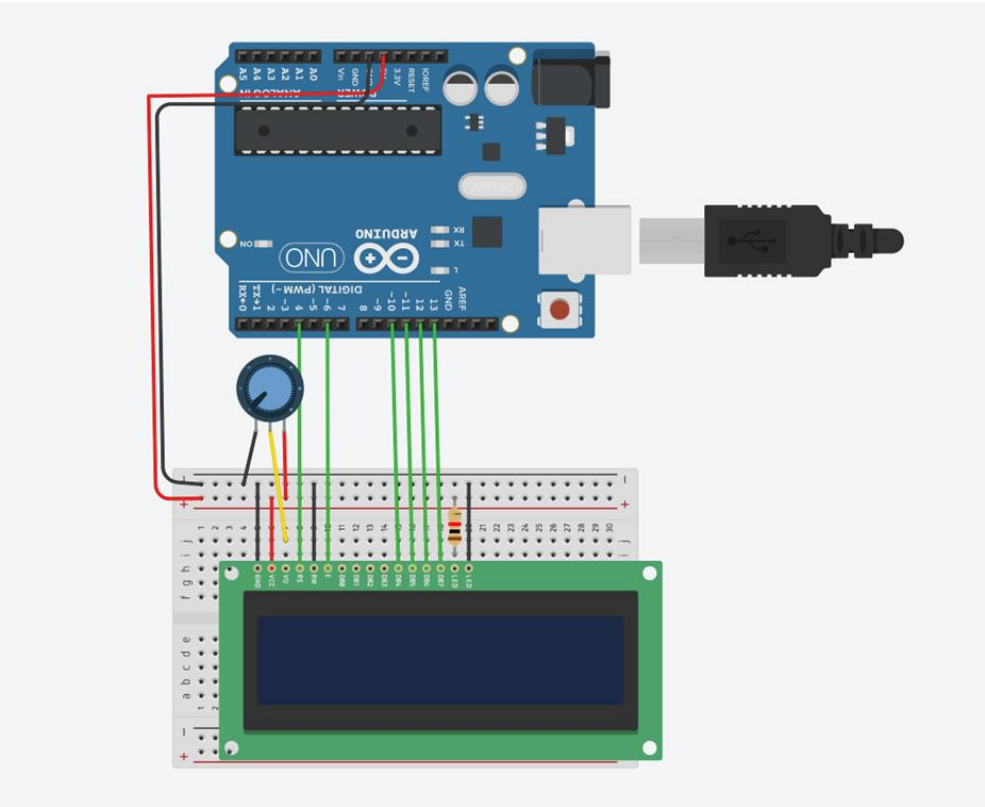

Figure 2. Circuit Diagram of Experiment 2 and 3.

LCD is used here to display the sum of an array, calculated using for statement. Then students are asked to calculate and display factorial of a number using while statement first, then calculate and display sum of 2 arrays.

2.1.3 Experiment 3: pointers

Pointers is one the most difficult subject to understand for students. LCD is used here to display value and address of a pointer. Pointer arithmetic is used to display elements of an array and their addresses. Students asked to write a function that takes 3 arrays and one integer for size of arrays and write the sum of first 2 arrays into the 3rd array.

2.2 Results

During the laboratory sessions, preparing the circuit according to the diagram might be time consuming because of the inexperience of the students who have not yet took the Circuit Theory course. For this reason, an instructional module experimental card is designed to use in programming laboratory sessions. In this way, a platform is created in which students could perform their algorithms and use various components.

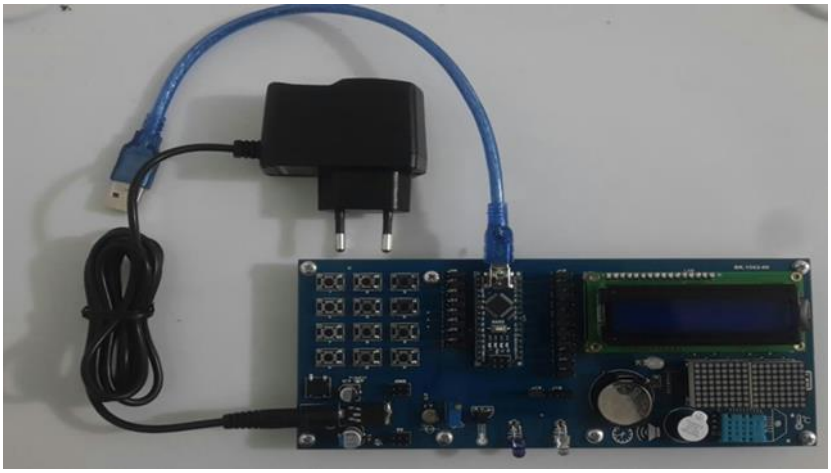

Figure 2. Instructional module experimental card.

A buzzer, humidity sensor, temperature sensor, LCD, Dot matrix display, buttons, RGB LED, potentiometer, LDR and a keypad are the components on the card. Students have the opportunity to physically experiment these components with programming Arduino.

3 CONCLUSIONS

In this study instructional modules are designed with Arduino platform to make physical sampling of classical programming course content for students. Laboratory sessions of Algorithms and Programming course are supported with these modules. Three of the modules are given as examples. The instructional module is created on an experimental platform with the microprocessor-based card, Arduino. With this platform, visual and physical experiments are brought into use of students in the laboratory. The physical counterparts of the programming knowledge are provided to the students to learn easier and have an insight for programming.

As a result of laboratory session experience, an instructional module experimental card is designed to shorten the time students spend with setting the physical module. With the designed instructional module card, the experiments in the laboratory sessions will become varied.

ACKNOWLEDGEMENTS

The instructional module experimental card designed here financially is supported by Gebze Technical University with BAP 2017-A-105-58 project.

REFERENCES

[1] B. Furman, E. Wertz, “A first course in computer programming for mechanical engineers,” Proceedings of 2010 IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, Qingdao, ShanDong, 70-75, 2010.

[2] H. Qin, “Teaching computational thinking through bioinformatics to biology students,” ACM SIGCSE Bulletin, vol. 41, no. 1, 188–191, 2009.

[3] J. Wells, R. M. Barry, A. Spence, “Using Video Tutorials as a Carrot-and-Stick Approach to Learning,” IEEE Transactions on Education, vol. 55, 453–458, 2012.

[4] K. E. Merrick, “An Empirical Evaluation of Puzzle-Based Learning as an Interest Approach for Teaching Introductory Computer Science,” IEEE Transactions on Education, vol. 53, 677–680, 2010.

[5] A. Ruthmann, J. M. Heines, G. R. Greher, P. Laidler, C. Saulters II, “Teaching computational thinking through musical live coding in scratch,” Proceedings of the 41st ACM technical symposium on Computer science education, 351–355, 2010.

[6] M. A. Rubio, R. Romero-Zaliz, C. Mañoso, A. P. de Madrid, “Enhancing an introductory programming course with physical computing modules,” 2014 IEEE Frontiers in Education Conference (FIE) Proceedings, Madrid, 1-8, 2014.

[7] J. Grasel, W. Vonnegut, Z. Dodds, “Bitwise Biology: Crossdisciplinary Physical Computing Atop the Arduino,” 2010 AAAI Spring Symposium Series, 2010.