

# Using Robot Platforms to Enhance Concept Learning in Introductory Computer Science Courses

Colleen Elizabeth van Lent

Computer Engineering and Computer Science Department  
California State University Long Beach  
1250 Bellflower Blvd  
Long Beach, CA 90045  
[vanlent@cecs.csulb.edu](mailto:vanlent@cecs.csulb.edu)

## Abstract

A common pedagogical challenge in introductory courses is how to engage student interest without exceeding the limits of what can or should be expected from students learning basic concepts. Standard assignments are typically artificial, and not readily applicable to larger problems. One way to provide educators and students with a “real world” setting is to incorporate a robotic platform. This project will test the hypothesis that embedding robotic in non-traditional venues will motivate students to understand the underlying problem-solving concepts and develop a deeper understanding of programming constructs.

## Statement of Need

A common problem in introductory courses offered at college and university campuses is how to engage student interest without exceeding the limits of what can or should be expected from students learning the rudimentary concepts of a given field. This is true in CECS 174: Programming and Problem Solving I, an introductory course offered by the Computer Engineering and Computer Science Department at California State University, Long Beach. This course suffers from a common problem within computer science education, a student belief that earning a degree in the field of computer science simply involves learning one, possibly two programming languages. Therefore, instead of striving to fully grasp the basic concepts of computer science, students mass produce code to get each specific assignment done. This practice fails to nurture the problem-solving skills necessary for ongoing career success, regardless of their field of study.

To further exacerbate this problem, it is extremely difficult for instructors to develop interesting assignments that only use concepts within the scope of introductory programming and problem solving. Since educators are bound by this constraint, few assignments culminate in anything more than a display of alphanumeric values on a computer monitor. Because problem sets are typically artificial, students do not always recognize that these basic concepts have applications to larger problems.

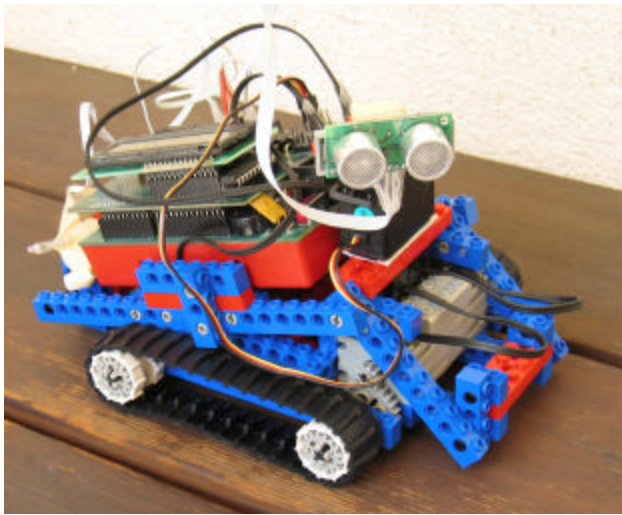
Obviously new approaches to accessing programming assignments are necessary. Hopefully, platforms that can sense and move (as opposed to textual I/O used with typical programming assignments) will provide a more active learning environment and promote student interest.

## Project Goals, Objectives, and Predicted Outcomes

CECS 174 is a Freshman level introductory course offered to both CS majors and non-majors. The course is divided into two modules, a 50 minutes lecture followed by a 75 minute lab. The labs are intended to provide students with the opportunity to apply the principles covered in lecture. The goal of revamping the 174 course is to create a classroom environment that will be conducive to demonstrating how simple concepts can be used to tackle large-scale problems. In order to obtain this goal the three objectives are:

- 1) Develop a new set of introductory programming assignments that will challenge the students to use problem-solving skills to solve real-world problems using a scientific design. These assignments must be applicable to both traditional and robotic-platform enhanced learning environments.
- 2) Develop tools that will allow other instructors to incorporate the platforms into their classrooms without the need to modify their students' work.
- 3) Research the benefits and potential pitfalls of incorporating robotics into an already challenging classroom experience.

This course requires a small, mobile robot lab to provide students with a novel, team-building environment in which to learn. The robots will provide an additional model to view the importance and applicability of introductory concepts. Students will need to learn that answers that are not well thought out might execute correctly on a computer, but fail in a real world environment. The end result will hopefully be improved problem-solving skills and increased student retention in the later courses.



Additionally, it is hoped that other instructors will also benefit and contribute to the outcome of this work. Once educators are instructed on how to use the tools to enhance their own classroom experience, they can use my concept outline as a platform for their own courses. The results of the performance comparison can be used for the comparison of pedagogical techniques among instructors of the same course.

### Description of Activities

Currently two sections of CECS 174 (Programming and Problem Solving I) are being taught by the same instructor. In one section the set of programming assignments requires common keyboard input/monitor output. In the other section the same concepts are covered, but in addition to the visual display, the code embedded onto a robotic platform using Interactive C [1]. For instance, while one section may implement a loop that counts to 10, the other will implement a loop that spins the robot. In another assignment the first group may halt their program when a certain number is entered, and the second group will halt the robot when a bumper is hit. To date the instructor has :

- 1) Identified programming concepts within CECS 174 that can be translated into engaging programming assignments for applied and non-applied courses.
- 2) Created robot platforms. The robot chassis were designed using information from The Art of Lego [1]. The processor used on each is a Handy Board, which is based on the 52-pin Motorola MC68HC11 processor, and includes 32K of battery-backed static RAM, four outputs for DC motors, a connector system that allows active sensors to be individually plugged into the board, an LCD screen, and an integrated, rechargeable battery pack [2]. In

addition, each robot is equipped with two motors and multiple, removable sensors. One of the robots built for the class appears in Figure 1.

- 3) Outlined differences between the C++ programming language used by all 174 courses, and the Interactive C language [3] used to program the robots.

It is important to note that none of the planned activities involves changing 174 into a robotics course; the robots are used to create a more immersive environment. The students learn about the importance of functionality, passing proper inputs, and receiving expected output. An initial period was needed to introduce students to the minimal concepts of programming needed to write program: variables, expressions, operators, program syntax, and program structure. The robot labs are intended to begin concurrently with the introduction of functions. Simple functions such as `motor(x,y)`, `ao()`, `sleep()`, `digital(x)`, and `beep()` can produce interesting programs such as:

- traveling in a square.
- detecting whether an object is light or dark
- line following;
- responding to stimuli.

Some additional concepts that were not the intended focus of the lab, but still addressed were infinite loops, constants, and the scientific approach to testing.

### Budget

An itemized budget for purchasing the processors, sensors, and building materials for eleven robots is provided in the Table 1. The original plan was to work in ten groups of three and have enough additional equipment for replacement pieces and one robot for the instructor. In the end, the dynamics of four students to a group was more effective. It also allowed for the numerous “breakages” when a robot would fall apart. For my purposes, I had an overabundance of Legos™ and motors. The CMU camera was purchased for use in more advanced courses.

### Assessment

The expected outcome of incorporating the robot platforms is that students will have a better grasp of CS concepts and hopefully, become more interested in CS as a field of study rather than as a programming language. This can be evaluated in four ways: programming competence, basic concept competence, student surveys, and retention.

Three common exams will be given to both sections of 174 and each exam will have a programming and written component. In addition, all students enrolled in the ten sections of 174 will be required to through a

Equipment description	Price
Prepackaged robot kit with motors/sensors	~1125
Four (4) prepackaged Lego bags (~1800 Pieces of Lego/bag)	~1200
Ten (10) Handy Board (HB) processors, expansion boards, and Interface HW	~3500
CMU Camera modified for use with HB	~120
Thirty (30) Black Gear Motors	~650
Ten (10) Servo Motors	~150
Eighty (80) Digital Sensors	~750
Eighty (80) Analog Sensors	~850
Ten (10) Sonar Sensors	~350
Total	~\$8695

Table 1: Equipment needed

comprehensive. Therefore, I will be able to compare the results of the students who used the robot platform against not only my other section of 174, but all sections of 174 offered in the same term. The student survey will ask the students to assess the quality of the twice weekly lab periods based on their usefulness of demonstrating the introductory programming course objectives. The final form of evaluation will look at whether the physical platform increased interest in the follow-up course by comparing the percentage of students who choose to continue in CECS from each course. It will also be interesting to see how students from varying backgrounds and genders respond to this approach. A benefit of conducting this study early in the CECS sequence is that there is still a wide range of students enrolled. And while the students in just one class can not represent every demographic, their results may lead to new ways to address retention. However, the approval to obtain personal information is still unattained.

## References

- [1] Interactive C; <http://www.newtonlabs.com/ic/>
- [2] Martin, Fred G. *The Art of Lego Design*, 1985. The Robotics Practitioner: The Journal for Robot Builders, volume 1, number 1, <http://web.media.mit.edu/~fredm/papers/artoflego/cliche.html>
- [3] HandyBoard; <http://handyboard.com/>