Computer Science Outreach Curriculum Unit — Robot Programming

Lego Mindstorms "Can Do Challenge"

By Stacey Fornstrom

Target Audience

This lesson is for Junior High and High School students with general knowledge of how to use a computer.

The lesson that I am attaching is one that I use with 9th graders. However, the beauty of using Lego Mindstorms is that the scope of the lessons and projects can be adjusted from very simple to quite complex. My lesson would fall into a medium-difficulty range because I introduce a lot of beginning computer programming concepts with my lessons. At the high-end range, interfaces exist that allow students to program the robots with more complex languages like Java.

Computer-science concepts found in this lesson

Lego Mindstorms is an excellent way to introduce students to a variety of concepts and skills, including: introductory computer programming, object-based programming, event-based programming, computer hardware, project design and planning, group work, engineering, and working with their hands.

Specific computer-science concepts found in this lesson include:

- Algorithmic thinking
- Objects – an occurrence of a class type that has properties and methods.
- Hardware – the physical part of a computer
- Software – instructions the computer follows
- Programming – creating saved instructions
- Steps to writing program
    1) define output
    2) develop logic to get to output
    3) write program
- Flowchart – graphical representation of a computer program.
- Machine language
    - series of 0's and 1's that provide instructions for computer
- High-level language

  - Makes it easier for humans to communicate with computers. We don't have to write program as 0's and 1's.
- Input – information given to a computer
- Output – information returned from a computer
- Events – something that happens in a program; i.e. mouse clicked
- Conditional Statements – If/Then, decisions made by a computer
- Loops – instructions executed many times

Necessary Resources

- "Lego Mindstorms" robot kits - contains all necessary pieces and software that can be used with Windows. The cost is approximately $200.
- One computer for every two students. Mac or PC's may be used if the Robolab software is purchased. A site license for Robolab costs $230. PC's must be used if the software that ships with the Mindstorms robot kit is used.
- For the "Can Do Challenge" I fill seven empty pop cans with sand and duct tape the top hole. I made a white board with an oval ring that is outlined by dark tape.
- There are additional curriculum resources and examples at www.lego.com/education.

Time Needed

The scope of the project can vary, based upon the needs of the students. An introductory lesson can be compressed into a 2-hour session. More complex projects might take 2 weeks (or more) of 1 hour sessions.

For the "Can Do Challenge," I allocate approximately two weeks of 50 minute class sessions.

Related Disciplines

Other areas this unit can tie into include: Science, Technology, Industrial Technology, Math, or Engineering. In my lessons I focus on skills related to teamwork, project planning, project management, and communication - including class presentations and written reports.

Students’ Motivation

I have framed my lesson as a problem by using the “Can Do Challenge.” I provide the following introduction to students. Details are in the file named: “RobotProgrammingRubric.doc”.

Instructions for Students

Lego Robot “Can Do Challenge”

You will work in teams of two to complete the following challenge: design, build, program, test, debug, and document a Lego robot that can move soda cans out of an oval rink. Your team may use any robot design that you feel will work the best. Remember that the program needs to be able to work with your robot.

The rink is made from a table top and has blue tape for the boundary line. The start box for the robot is outside the blue tape and is a 25cm x 25cm house shape made of green tape. Your robot must fit inside the boundary of the start box. ***The robot cannot expand to a size larger than the start box at any time!*** Your robot will have 2 minutes to move as many cans from the oval as possible. Their will be 3 competition sessions, and the number of cans moved out in the 3 sessions will be tallied to determine the champion team.

The design for your robot design must be approved by your teacher before you begin building the robot.

A tentative schedule, including items to be turned in by each team, is as follows:

Item	Resources	Due Date
Begin project by brainstorming as a group and documenting project plans.	Use Word and any other software you like to document ideas from the brainstorming session.	**Weds. 10/29**
Group shares ideas for robot design		**Thurs. 10/30**
Preliminary Project Plan – Vision, Plan, and Implement sections. (20 Points)	“Project Management Model Template.dot” Project grading rubric	**Mon. 11/3**
Robot built	Lego Mindstorms robot kit	**Tues. 11/4**
Robot program complete	Robolab programming software	**Weds. 11/5**
1st round of testing	Robots, program, test board	**Weds. 11/5**
Modifications based on round 1 tests		**Thurs. 11/6**
2nd and 3rd rounds of testing		**Fri. 11/7**

The following items should be considered in designing and programming the robot:

- Designing the robot – use the “Constructopedia” book that is in your Lego kit for design ideas. You may also use the Internet to search for designs. Use your time wisely because it will not be unlimited.
- Moving and turning the robot – use the light sensor to keep the robot within the oval rink.
- Feeling the cans – consider using touch sensors to determine when the robot is in contact with a can.
- Moving the cans out of the rink – the best designs will have a mechanism that is able to grab or control a can so that the robot is able to move it.

Rules for working with Robots:

- Robots must be run on the floor, never on tables.
- No robot demolition derbies.
- Turn off the RCX when it is not being used.
- Have questions ready for the start of each class period so that we may review them as a class.

Introductory Material

Before introducing the “Can Do Challenge”, I spend one week teaching how to build and program the Lego Robots. I use the “Robolab Team Challenge Set” for curriculum, but teachers can design their own activities. Robolab has several curriculum resources, which may be found at: www.lego.com/education.

The primary purpose of using this curriculum is to acquaint students with the robots, the software, and encourage them to explore. My overriding theme is that they won’t break anything, so experiment!

The introductory week has three assignments. In the first assignment we build a robot called the Acrobat. The second assignment directs students to experiment with the built-in programs that come with Robolab and analyze and document the results. The third assignment shows students how to program the robot. Students create five programs, with each program becoming more complex.

I demonstrate techniques for students to use in each of the introductory assignments. Notes for teachers to use in the demonstrations are in the file named “RobotProgrammingNotes.doc”.

Main Curriculum Unit Body

The main curriculum consists of teaching the content included in the curriculum sections of this document. I have attached sample lesson plans in the file named "RobotProgrammingLectureNotes.doc". I try to reinforce computer science concepts during each class session and also provide students with time to create their own projects.

Extensions

There are several extensions that may be incorporated into the Lego Mindstorms curriculum. The first is visible in the challenges. Students can create more complex robot designs or more complex programs.

The robots may also be programmed with many different languages, including Java, Visual Basic, Ada, and others.

Assessment

I use the attached rubric to grade the "Can Do Challenge" project, in the file called "RobotProgrammingRubric.doc". Note that the grading focuses on the process and not the results. Students can receive a good grade if they design and follow a complete plan, even if their robot is not able to move any cans from the ring.

The robots make it easy for students to analyze results of their own programs by asking themselves "is the robot doing what I expected it would?" If the answer is no, what should I change? I believe that this is the strength of using the robots. It makes it easy for students to analyze their own results.