Lesson Title	Functions Challenge Follow-Up w/ Line Counting
Length of Lesson	3 Days
Created By	Dustin Spayde
Subject	Robotics
Grade Level	11-12
State Standards	
DOK Level	DOK 4
DOK Application	Design, Create, Apply Concepts, Analyze, Critique, Connect
National Standards	9-12: A(Inquiry), E (technology)
Graduate Research Element	Developing Automated Systems, Programming

Student Learning Goal:

National Science Education Standards of Content 9-12
A (Inquiry): Identify questions and concepts that guide scientific investigations.
E (Science and Technology): Abilities of technological design: propose designs and choose between possible solutions, implement a proposed solution, evaluate the solution and its consequences, communicate the problem, process, and solution; Understanding about science and technology

Materials Needed (supplies, hand-outs, resources): A Lego Mindstoms NXT kit per 5 students, Access to computers (one for each group) with USB ports and the RobotC software (or other compatible language) installed on each, black tape.

Lesson Performance Task/Assessment:
Students will be broken up into teams (groups of 5 for my class). Each team will be given a kit and told to build a robot that can complete the challenge. The challenge will be divided into one task for each team member (in my case the challenge was broken into 5 tasks). **If you are using this as a follow-up to a previous function challenge make sure that no student has the same task as before.** Most of these tasks should be custom functions programmed by the students (such as "Move Forward & Backward" or "Compass Turn"). Each of these functions should use variables that allow the function to be recalled by another program. These other programs should be the remaining tasks for each group (my class consisted of 2 basic functions, and 3 higher level programs for each team. The higher level programs are only allowed to use the basic functions created by their other team members to complete the challenge).

Grading was based on three items:
Robot design (group grade/individual participation grade)
Individual task grade
Challenge grade (group completion grade)

Lesson Relevance to Performance Task and Students:
An automated vehicle such as this could easy be found in many factories and ports around the world. Developing its navigation system is an applicable task for a many engineering fields. Designing, programming, and troubleshooting a robot design where each team member is responsible for one aspect of a project, is a very realistic scenario for actual engineering work.

Anticipatory Set/Capture Interest:
Discuss how today we are going to make programming much easier. Explain we are going to use a new method to move the robot.

Guided Practice:
Day One: Lecture on how line counting and variables work. Make sure to have an example so that students can see the flow of the program. Then break students into their teams and detail the challenge.

Independent Practice:
Day 1: Students must first build a robot

Day 2-3: Students must program their functions and work together to complete the overall challenge.

Remediation and/or Enrichment:
Remediation: individual IEP; partner help throughout lesson; shorten parts of assignment; focus upon smaller elements of the process

Enrichment/Extension:

> After teams have had time to build their robots, then break up the teams seating arrangement into departments. Each department groups all of the students in the class who have the same task. This is done to force a more likely work scenario. In this configuration teams members will only discuss issues with each other's programs at the testing location. However departments of students may openly help each other. This is a very realistic scenario for actual engineering projects and forces students to overcome many logistic obstacles.

Check(s) for Understanding:

Day One: Do all groups have the basics of their robot design worked out?
Day Two: Have all students begun tested their programs? How can you improve your robot's performance?
Day Three: All groups should have the completed the challenge using only their functions?

Closure:

Ask which design is the best and why?

Possible Alternate Subject Integrations:

*Math – can manipulate mathematical expressions to isolate needed variables

*Programming – Basic logic and algorithm models

Teacher Notes: below is a picture of the course I designed for this lesson. It requires the robot to start in the circle, and then use line counting to go down one of the alleys and use an arm to collect an item. The robot must then reverse, using line counting again, back to the circle. Finally the robot must turn using the compass sensor to find the correct heading and deposit the item in its respective target area. This is done 3 times, one for each alley.