Getting to know your Robot

Introduction:

There are examples of robots all around us in our everyday lives. Washing machines, air conditioners, cars… all contain robotic devices. Robots are no longer the domain of science fiction. Microprocessors are now extraordinarily cheap and many of them can be reprogrammed using computer software. There are two levels of programming on such microprocessors; firmware and software. The robot you will use in this investigation has been programmed (firmware) to accept commands from a graphical calculator. You will use your calculator to write software for the robot's microprocessor. This investigation explores the commands you can use to control the robot's movement.

Equipment:

TI-83/TI-83Plus/TI-84Plus graphical calculator
Small Robot
Unit to Unit calculator cable (on robot)

Getting Started:

- Connect the calculator to the robot via the communication cable.

- Switch both the calculator and robot on.

- Press the PRGM button on the calculator and start a new program.

- Name the program "ROBOT1"
  Note: You will need to press the ALPHA button to access '1' after typing 'robot'.

- Press the PRGM button, move across to the I/O (input / output) menu and select **Send(** This command transfers instructions that you have typed into the robots microprocessor, the software the robot reads and interprets in order to move.

- In the send command enter the values: {122,100 }. Close the brackets on the send command and press ENTER to complete the instruction. (Refer calculator screen shot)

> ✐ Any time a **set** of numbers is used on the calculator, use { } to enclose the numbers. This is how the calculator knows that a **set** of numbers is being used.

- Press the PRGM button, move across to the I/O (input / output) menu. The **Get(** command forces the robot to execute the most recent instruction set and also retrieves information from the robot's microprocessor. The retrieved information will not be used in this investigation.

- Type in 'R' using the ALPHA command and close the brackets. Your first program is finished!

- Press 2nd MODE to quit the program editor.

- Make sure your robot is on a large, flat, level surface free of any obstacles. (Low profile carpet, vinyl… works best)

- Press the ⌈PRGM⌉ button, select the "ROBOT1" program, press ⌈ENTER⌉ to select the program, then ⌈ENTER⌉ to run the program. Observe what your robot does!

Question - 1

Write down a brief description of what your robot did using the instructions: {122,100 }.

- Once the robot has finished moving, press the ⌈PRGM⌉ button on the calculator, highlight your ROBOT1 program, select **edit** and press ⌈ENTER⌉. Change the instructions in the send command to: {122,200 }.

```
PROGRAM:ROBOT1
:Send({122,200})

:Get(R)
:
```

- Run the ROBOT1 program and observed the robot's movement once again.

Question 2:

What did your calculator robot do this time? What is the purpose of the second number in the **Send(** command?   (ie: Compare Send({122,100}) with Send({122,200})

Question 3:

Change the instruction to {122, 250}; run the ROBOT1 program once again and observe your robot's movement. How does the second number in the instruction set affect the robot's movement?

> In the next activity the second command in the **Send(** instruction will be investigated (calibrated) further. Check your answer to question 3 with your teacher before proceeding any further with this investigation.

Exploring the Send(command.

The robot can move backwards or forwards, turn left or right, respond to its bump sensor and make decisions based on this input via intelligent programming. The next stage of this investigation focuses on the first instruction in the **Send(** command.

- Edit the ROBOT1 program and change the **Send(** instruction to: Send({121,100})

- Quit the program; place your robot on a large, flat, level surface free of any obstacles. Run the ROBOT1 program and observe very carefully the robot's movement. Pay particularly close attention to the wheels. You may need to run the program a couple of times to notice what is happening.

```
PROGRAM:ROBOT1
:Send({121,100})

:Get(R)
:█
```

Question 4:

Write a description of what your robot did using the instructions: {121,100 }. In your description include information about how your robot achieved this manoeuvre.

> Mathematics is not simply about repeatedly performing algorithms in order to determine unique solutions to specific problems that have little or no resemblance to real world situations. Mathematics and mathematicians are valued for their ability to think logically, observe patterns and make accurate predictions or conjectures based on knowledge or information already obtained. So far you have observed two aspects of the **Send(** command. You will now be required to make a prediction based on your experiences.

Question 5:

It is now time to combine your knowledge of the commands used so far in order to make predictions. Write down what do you think will happen if you type in the send instruction with the set of numbers: {121, 200}. Change the program and then run it. Observe what the robot does, this is how you correct your answer. This is a little different that looking in the back of a textbook for the answer!

Further exploration of the Send(command.

You should now be confident changing the program contents without detailed instructions. The **Send(** command is limited to a range of values. Use the table below to help you explore some more aspects of this command. Copy the table and write your observations next to each instruction set. Your instructions need to be detailed observations not only of the physical movement, but also how the movement was achieved.

Instruction	Observation
Send({120,100})	
Send({110,100})	
Send({102,100})	
Send({101,100})	
Send({112,100})	
Send({100,100})	

Question 6:

Make a prediction for the command Send({111,100}), test your prediction and explain how this may be a useful function for a robot to perform.

Mathematicians and scientists often collect information often referred to as research. The accurate documentation of this information is critical. The information you have collected in this investigation will be used in future investigations. It is therefore REALLY important to ensure your information is correct, accurate and sufficiently detailed to ensure future investigations are successful.