**//Line Tracking Test**

**//Name:**

//Advanced Robotics

//Date:

1.) The first line tracking program we wrote used while statements inside of if statements. Why can nested loops cause a problem in a program?

*The program can get stuck in the while loop.*

*The program is not able to exit until the while loop condition is met which makes it difficult to test multiple conditions at once.*

2.) Answer the questions about the following segment of code:



What will the robot do if the light sensor reads a value of 64?

*Motor C will turn on. It will stay on until the reading goes below 45.*

What if it reads a value of 45?

*Motor B will turn on. Notice this is the equal case. It makes the “if” statement false. It will stay on until the reading goes above 45.*

This program will only check the lightsensor value once. How can you make it run forever?

*Put the whole program into a “while” loop: while(true) or while (1==1)*

3.) The program below makes the robot:



a.) travel at a speed that varies continually based on the value of the sound sensor, for 1 second.

b.) travel at a set speed based of the initial value of the sound sensor, for ten seconds.

c.) travel at a speed that varies continually based on the value of the sound sensor, for 10 seconds.

d.) travel at a set speed based of the initial value of the sound sensor, for one second.

Explain why your answer is correct.

*B. This program takes the value of the sound sensor at lines 5 and 6. It then turns the motors on during that time, then the program goes to sleep for 10 seconds.*

4.) Explain, in terms of “values”, why the amount of sound you made affected how quickly the robot moved in the Speed Based on Volume program.

*The value of the sound sensor is used for the value of the motor power.*

5.) Errors! When we write programs it is part of life that we make mistakes. The compiler, ROBOTC, will not know what we are asking it to do when there are errors. Fix the program below to make it error free!

 

6.) Write a program that makes the robot go forward for 5 seconds, then turn around for 3 seconds. You may NOT use wait1Msec statements.



7.) Write a program that makes the robot track a line until the touch sensor is pressed.



8.) Write a program that makes the robot turn in a circle at different speeds depending on how much noise it senses.



9.) The following code tracks a line forever:



How would you change the code to make it track a line for exactly 10 seconds?



How would you change the program to make it track a line for exactly 9 rotations?

