

CIS 1.5 Spring 2007 Lab I, Part 2

Instructions

- This is the second part of the first homework/lab assignment for CIS 1.5. Read the first part of the assignment for complete instructions, due date and submission details.

1 A more sophisticated antworld

(2 points)

1. Follow the “How to use CodeBlocks” instructions to create a new project called **antworld**.
2. Get a copy of the antworld C++ code from Professor Parsons, also given in Appendix A, and copy it into the **antworld** project.

Make sure you change the comment to include your name.

Compile, Build and Run your program.

3. Now, we will make the antworld 11 squares in both directions, so that the x and y values must be between 0 and 10. For each of the exercises below, you will have to repeatedly edit the antworld file, then compile, build and run it, until it does as described.

- (a) Add error checking in the **moveNorth**, **moveSouth**, **moveEast**, **moveWest** methods so that the ant does not go out of the world (in other words so that x and y don't go above 10 or below 0). To do this you will need to add a check before you increment or decrement x and y .

(1 point)

- (b) Now make it so that if the ant moves north off the north side of the world, it appears at the south side. Write a message to the screen when this happens.

Make sure that your changes do not alter the error checking you just introduced.

(1 point)

- (c) Extend your answer to the previous question so that when the ant moves south off the south side of the world it appears on the north side, when it moves west off the west side of the world it appears on the east side, and when it moves east off the east side it reappears on the west side.

Write a message to the screen when each of these things occurs.

(1 point)

- (d) Add a new behavior **moveNorthToEdge()** that causes the ant to move forward until it gets to the North edge of the world and then stop.

Add similar behaviors **moveSouthToEdge()**, **moveEastToEdge()**, and **moveWestToEdge()** which move the ant to the edge of the world in the indicated direction.

Extend the set of user commands so that the user can tell the ant to move forward until it gets to the edge of the world. Use the following letters:

Q	Quit the program
n	move the ant north one step
s	move the ant south one step
e	move the ant east one step
w	move the ant west one step
N	move the ant north to the edge of the world
S	move the ant south to the edge of the world
E	move the ant east to the edge of the world
W	move the ant west to the edge of the world

(1 point)

4. Save this last version of the code (the one with the new commands). This version of the project should be submitted as the second part of your first homework/lab assignment.

This version of the code should contain *all* of the changes you have made during these exercises — you will only get credit for the things that I can see you have done. You can use comments to make sure that I see all these things.

Appendix A

```
//-----  
//  
// A more interactive version of the antworld.  
//  
// Simon Parsons  
// February 8th 2007  
  
//  
// Include C++ library definitions  
  
#include <iostream>  
using namespace std;  
  
//  
// Declare variables  
  
int x; // robot's x position  
int y; // robot's y position  
char c; // user's input  
bool q; // does user want to quit?  
  
//  
// Declare methods  
  
void displayPosition()  
{  
    cout << "the ant is at location (";  
    cout << x;  
    cout << ",";  
    cout << y;  
    cout << ")\n";  
}  
  
void moveNorth()  
{  
    cout << "moving North...\n";  
    y = y + 1;  
}  
  
void moveSouth()  
{  
    cout << "moving South...\n";  
    y = y - 1;  
}  
  
void moveWest()  
{  
    cout << "moving West...\n";  
    x = x - 1;  
}  
  
void moveEast()  
{  
    cout << "moving East...\n";  
    x = x + 1;  
}
```

```

//
// Define main method

int main()

{
    x = 0;                // Set variables
    y = 0;
    q = false;

    while ( q==false )    // We keep doing this bit
    {

        // Get input from user

        cout << "Which way should the ant move (enter N,S,E,W or Q)? ";
        cin >> c;
        cout << "You entered: ";
        cout << c;
        cout << "\n";

        // Depending on what the user entered, move the right way

        if ( c=='N' )
        {
            moveNorth();
            displayPosition();
        }

        if ( c=='S' )
        {
            moveSouth();
            displayPosition();
        }

        if ( c=='E' )
        {
            moveEast();
            displayPosition();
        }

        if ( c=='W' )
        {
            moveWest();
            displayPosition();
        }

        // If the user entered Q, set up so that we finish.

        if ( c=='Q' )
        {
            q = true;
        }
    }

    cout << "Time to go!";
    return 0;
}

```