

Programming Project #7

EGRE245 Spring 2015

Turtle Graphics

1 Overview

The Logo programming language made the concept of *turtle graphics* famous. Imagine a mechanical turtle that walks around a board under the control of a C program. The turtle holds a pen in one of two positions, up or down. While the pen is down the turtle traces out lines or shapes as it moves; while the pen is up, the turtle moves about freely without writing anything. In this program you'll simulate the operation of the turtle and create a computerized sketchpad as well.

Use a 25×25 board which initially has no marks on it. The turtle always starts at position (0,0) (row first, then column; position (0,0) is the upper left corner of the board) facing east with its pen up. The turtle will also always be facing one of four directions – north, south, east, or west (with north being up on the screen, east to the right, etc.). Your turtle will then be given a set of commands (as integers) that will control its movement and drawing on the board. Here are the possible commands:

Command	Meaning
1	pen up
2	pen down
3	turn right 90°
4	turn left 90°
5 x	move forward x spaces
6	output the board
9	end of data

Note that the turtle can only move to the edge of the board, e.g. if a move command calls for moving 5 spaces and after moving only 3 spaces the turtle is at the edge of the board, it stops and does not complete the remaining 2 moves. Also note that when a pen down command is given the current location of the turtle should be marked.

2 Program Design

You must subdivide your program so that each command (pen up, pen down, turn right, turn left, move forward, and output the board) is executed in your code by calling a separate function. The *only* global variables you are allowed are for the current board position of the turtle, whether the pen is up or down, which direction the turtle is facing, and the board (a two dimensional array).

3 Input & Output

All input should come from standard input (which normally would come from a file redirected on the command line). You may assume that the input is correct. Output should be directed to standard output and only be produced when the output the board command is given. You must output your board exactly as it appears in the sample run below.

4 Sample Run

The sample run below was produced with the following data:

```
5 2
3
5 2
4
2
5 5
3
5 5
3
5 5
3
6
9
```

```

Terminal — tcsh — 68x36
liberty:~/docs/classes/egre245/projSolns/% gcc proj7Spr15.c
liberty:~/docs/classes/egre245/projSolns/% a.out < turtle.dat
EGRE245 Proj. #6 - D. Resler

      1               2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4
+-----+-----+
0|                                           |0
1|                                           |1
2|   * * * * *   |2
3|   *           *   |3
4|   *           *   |4
5|   *           *   |5
6|   *           *   |6
7|   * * * * *   |7
8|                                           |8
9|                                           |9
10|                                          |10
11|                                          |11
12|                                          |12
13|                                          |13
14|                                          |14
15|                                          |15
16|                                          |16
17|                                          |17
18|                                          |18
19|                                          |19
20|                                          |20
21|                                          |21
22|                                          |22
23|                                          |23
24|                                          |24
+-----+-----+
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4
      1               2
liberty:~/docs/classes/egre245/projSolns/%

```

5 Deliverables

Name your source code file `proj7XXXX.c` where `XXXX` is the last 4 digits of your student id number. For example, if your student id number is V12345678, your file will be named `proj75678.c`. Document your code in the manner discussed in class, and submit your project in the usual way via the web using the link off of the class web page.

Due date: Thursday, April 16