Electrical & Computer Engineering Department, University of Maryland, College Park

Spring 2007 **ENEE114** Dr. Gang Qu

Project 3: Easter Egg Hunt

Posted: Monday April 2, 2007.
Described: Tuesday April 3, 2007.
Due: 11:59PM, Monday April 23, 2007.

Project Objective:

1. master the design of simple user interface to interact with the user.
2. get familiar with multi-dimensional array.
3. learn to solve problems by recursion.
4. get more familiar with selection of data structure to store input data.
5. get familiar with handling interactive input with simple user interface.

Project Description:

In this project, you will implement a program for Easter egg hunt in an 11x19 maze. Your program should (1) read in from an input file the maze information; (2) show the egg hunter's current location; and (3) repetitively moves the egg hunter to the direction of his chooses until the egg is found, or the hunter gets lost and asks for help, or the hunter quits. Your program should display the floor plan of the maze and a path to find the egg at the end.

A master program, in the format of executable file, and sample input files will be posted on the class web site.

Input

The input file has exactly 211 lines of the following format:

- The first line has two integers, **x1** and **y1**, specifying the x- and y-coordinators of the starting location of the egg hunter.
- The second line has two integers, **x2** and **y2**, specifying the x- and y-coordinators of the egg.
- Each of the rest lines has exactly six integers, **x3**, **y3**, **W**, **N**, **E**, **S**, specifying the 209 cells in the 11x19 maze. W, N, E, and S can be either 1 or 0, indicating whether there is a wall (1) or not (0) on the west, north, east, and south of the cell that has x- and y-coordinators x3 and y3. For example, 1 1 1 1 means a cell with walls on all four directions; 1 0 1 1 means a cell with 3 walls on the West, East, and South; 1 1 0 0 means a cell with 2 walls on the West and North; and 0 0 1 1 means a cell with 2 walls on the East and South.

During the egg hunter, your program will read in the direction that the egg hunter wants to move to from the monitor.

Output

This project requires you write your output both to the monitor and two output files.

At the start of the egg hunt and after the egg hunter enters a legal move, you should print out the wall structure of the current cell. For example, the four cells in the input part, 1 1 1 1, 1 0 1 1, 1 1 0 0, and 0 0 1 1 should be described as:

```
 ---                 ---
| T |     | T |     | T        T |
 ---       ---                ---
```

where we use letter T to indicate the current location of the hunter, who always stands facing North (up). Therefore, West will be on his left, East on his right, and South is behind/down. A wall on North or South is represented by 3 consecutive –'s (minus sign) and a wall on the West or East will be represented by one | (vertical bar). Between the West and East walls, we have the constant pattern " T " (a space, a T, and a space).

Your program should provide the basic user interface to interact with the egg hunter. At any point, the hunter can enter "H" for help, "Q" for quit, "W" or "N" or "E" or "S" to move to the corresponding direction if there is no wall on that direction. Other input will be considered as illegal. See the master program for details.

The first output file is the floor plan of the maze that your program read in. You need to print out each cell in the format described above. Print "T" inside the starting cell, "G" inside the cell where the Easter egg hides, and leave other cells empty.

The second output file gives the path from the starting cell to the egg. This path is either found by the hunter or by your program (when the hunter asks for help). This output file should be empty should the hunter quits. The path is represented by a sequence of moving directions (W/N/E/S). For example, if from the starting cell, the hunter moves to the North once, then moves to the East 4 times and finds the egg, this output file should be exactly as follows

```
NEEEE
```

Project Requirements:

1. You must program using C under GLUE UNIX system and name your program **p3.c**.
2. Your program must be properly documented.
3. Submit your program **p3.c** electronically before the due time.
4. **IMPORTANT:** Your program's output, both to the output file and to the computer monitor, should be exactly the same as that produced by the master program.

Grading Criteria:

Correctness:	80%
Good coding style:	10%
Proper documentation:	10%
Late submission penalty:	-40% for the first 24 hours No submission will be accepted after the first 24 hours.