CSCD 240
HW 2

PROGRAM SPECIFICATIONS

You and your sister Gretel or you and your brother Hansel have been left in the forest. You must find your way out before the cannibalistic hag finds you, kills you and then boils your flesh.

The forest is a 20 x 25 2D array of characters. The characters have meaning:

T – a tree you can't move through it you must move around it

. – open path that you can move through

S – your starting position

F – your finishing position

H – the cannibalistic hag

How it works

Your array will be populated from a file. But you WILL NOT use FILE *. The program will be executed with ./hw2 < maze.txt. You will use scanf redirected from stdin. It will be well formed and we will live in the Happy part of Stuland.

You will ask the user for a direction to move, and you will read that direction in from the keyboard. Valid directions are entered as single characters N S E W. (You must ensure it is a valid character, case insensitive) Towards row 0 is north, towards row 19 is south, towards column 0 is west and towards column 24 is east. (If you really want to quit playing the user can type Quit (case insensitive))

Load the array, randomly place the starting point, randomly place the hag and display the forest. Here is the kicker. You can't display anything until you have entered that square. If the square contains a tree then you can display the tree after you have tried to move but could not. This means when you first start the only thing that will be displayed is the S for the starting position. This will be displayed in its proper space.

For each turn you will display a menu with the following options (You must ensure range)

1) Move (This will prompt for a move direction display the results to the screen)
2) Display the maze
3) Peek ahead (Display all squares within one of your location temporarily (no diagonals))

As you move through the maze you will write your movements to the screen.

Your output will look similar to the following:

Starting position [0][24]

Moving W to [0][23] – succeeded

Moving W to [0][22] – failed Tree blocks path

Moving S to [1][23] – succeeded

…

Moving W to [19][0] – succeeded found the exit

NOTE: if you come across the hag, you die and that game ends.

This maze was loaded with the command ./hw2 < maze.txt, and then the hag and the starting point were placed.

Sample Maze

```
T . . . . . . . . . . . . . . . . . . . . . T . S
. . . . . . . . . . . . . . . . . . . . . . T . .
. . . . . . . . . . . . . . . . . . . . . . T . .
. . . . . . . . . . . . . . . . . . . . . . T . .
. . . . T . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . H . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . T . . . . . . . . . . . .
T . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . T . . . . . . . . . . . . . . . T . . . .
. . . . . . . . . . . . . . . T . . . . T . . . .
. . . . . . . . . . . . . . . . . . . . T . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . .
F . . . . . . . . . . . . . . . . . . . . . . . .
```

What is Displayed to the Screen?

You will display an U for an unknown area or where you have been. See below for the sample (this is not a run)

Here is the maze:

```
U U U U U U U U U U U U U U U U U U U U U U S
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
```

```
Direction to move: W
```

Here is the maze:

```
U U U U U U U U U U U U U U U U U U U U U . S
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
```

```
Direction to move: W

Move failed -- T
```

Here is the maze:

```
U U U U U U U U U U U U U U U U U U U U T . S
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
```

```
Direction to move: S
```

Here is the maze:

```
U U U U U U U U U U U U U U U U U U U U T . S
U U U U U U U U U U U U U U U U U U U U U . U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
```

```
Peek
```

Here is the maze:

```
U U U U U U U U U U U U U U U U U U U U T . S
U U U U U U U U U U U U U U U U U U U U T . .
U U U U U U U U U U U U U U U U U U U U U . U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
U U U U U U U U U U U U U U U U U U U U U U U
```

And so it goes until you find the exit or the hag boils your flesh. If you find the hag display a sad message and stop the current run.If the user finds the exit, display a congratulations message and exit the program.

CODING SPECIFICATIONS

- You will have a hard coded two-dimensional array.
- There will be limited processing in main. The only thing in main:
    - Basic setup
    - Menu loop
    - I have provided a main
        - you can't change my function calls
        - you can add some function calls
        - you can change the name of the header
        - you can't change my constants
        - you can add some basic code/variables
- You will use a 3 file format. The file that contains main will be named cscd240_s13_hw2Tester.c
- I don't care what you call your .h or your implementation file(s)
- In your functions you must pass the array as hardcoded 2 dimensional
- There is no need for terms like malloc, realloc or calloc so you are not allowed to use them.
- There is no need for recursion
- You will submit a makefile that allows the grader to compile your code and produces the executable named hw2
- You must guard against moving off the maze
- The hag can move every 3rd turn in a random direction.

TO TURN IN

A zip that contains:

- Your C files
- all input file(s)
- a makefile

Name your zip your lastname first letter of your first name hw2.zip (Example: steinershw2.zip)

GET STARTED ASAP