CS1010E Practical Assessment #3: Sweep (Bonus Level)

Tags & Categories

Tags:

Categories:

Related Tutorials

Task Content

Sweep More!

Topic Coverage

- Assignment and expressions
- Control statements
- Functions and procedures
- Arrays

Problem Description

This is a bonus level to the Minesweeper task.

Notice that the original Minesweeper clears neighbouring squares as long as the centre square has no mine. The example shown for clearing the minefield at location `(4,1)` resulted in the following:

```
.............................
.............................
.32223.......................
1100011......................
0000001......................
2210001......................
..31101......................
....111......................
.............................
.............................
.............................
.............................
.............................
.............................
.............................
.............................
```

However, it can be observed that given the number of mines on the minefield, we can flag out the precise locations of some of the mines. This is indicated by `f` below.

```
.............................
.............................
f32223f......................
1100011......................
0000001......................
2210001......................
ff31101......................
...f111......................
.............................
.............................
```

```
.............................
.............................
.............................
.............................
.............................
.............................
```

In turn, given these flags, we can confidently clear more neighbouring cells as shown below.

```
.............................
.............................
f32223f3.....................
11000112.....................
00000011.....................
2210001......................
ff31101......................
..4f111......................
...223.......................
.............................
.............................
.............................
.............................
.............................
.............................
.............................
```

By repeatedly, flagging and clearing, we can clear a much, much larger area than before. For example, using only location (4,1) at the start, the cleared area is

```
......f20012f100001ff101ff....
......f4122f2100001221013f3...
f32223f3f4f4200000000000124...
110001122fff10000000111001f...
0000001123f3100111001f21013...
2210001f11111111f21013f2002...
ff31101110001f223f2102f4211...
.f4f11110111112f22f1023ff11...
ff3223f202f20011122201f3211...
2211f3f202f2000012f21123212...
000112222223321001f3f113ff12...
0122101f2f2ff100123323ff421...
02ff322132434222112f3f3f4f11...
13f6ff201f2f2f3f33f31212112...
.....f21333122....320000013...
.....211ff1001....f1000001f...
```

with the remaining mines shown below.

```
..*...f20012f100001ff101ff....
*.**.*f4122f2100001221013f3...
f32223f3f4f4200000000000124*.*
110001122fff10000000111001f*.*
0000001123f3100111001f21013*..
2210001f11111111f21013f2002...
ff31101110001f223f2102f4211*..
*f4f11110111112f22f1023ff11...
ff3223f202f20011122201f3211...
2211f3f202f2000012f21123212*.*
000112222223321001f3f113ff12*..
0122101f2f2ff100123323ff421...
02ff322132434222112f3f3f4f11..*
13f6ff201f2f2f3f33f31212112*.*
*..**f21333122.*.*320000013**.
.....211ff1001*...f1000001f.**
```

Let's call this clearing technique smartClear™.

Task

Write a program that reads an `m`-by-`n` minefield comprising of values `9` (representing a mine) and `-1` (safe). The program then repeatedly reads locations `(r, c)` of cells in which to clear and outputs the minefield after smartClear™ing the cells. The program terminates in one of three ways:

- Clearing the location of a cell with a mine;
- Quitting by clearing an invalid location; or
- When all safe cells are cleared.

Finally, the program outputs the minefield showing the mines, flags, as well as the cells that were cleared.

Sample Run

The following is a sample run of the program. User input is underlined. Ensure that the last line of output is followed by a newline character.

```
$ ./a.out
3 4
-1 -1 -1  9
-1 -1 -1  9
-1 -1 -1 -1
1 2
0 0
002f
002f
0011
```

Save your program as `sweepbonus.c`.

Submission (Course)

Select course: CS1010E (2017/2018 Sem 1) - Programming Methodology ▼

Your Files:

SUBMIT (only .java, .c, .cpp and .h extensions allowed)

To submit multiple files, click on the Browse button, then select one or more files. The selected file(s) will be added to the upload queue. You can repeat this step to add more files. Check that you have all the files needed for your submission. Then click on the Submit button to upload your submission.