Project 2 : Maze Router for 2-terminal Nets

Problem Statement

Given a gridded routing region, and implement a 2D maze router to complete two-terminal routing problem. You can use any routing algorithm you know.

A sample is given in Figure 1. There are two terminals, and blockages. The blockage indicates where the router is not able to pass through.

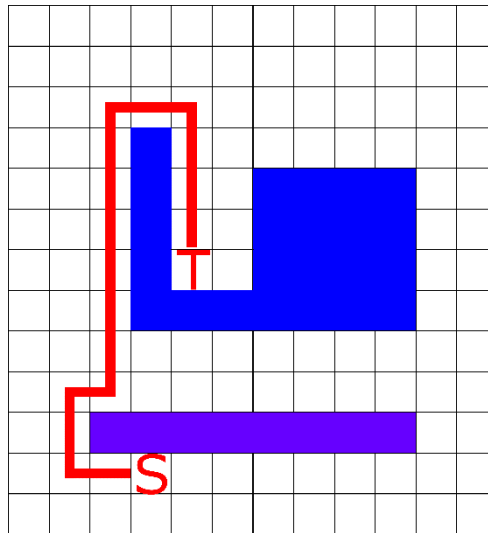Find a path from source to target, and minimized the number of wiring grids.

Figure 1

There will be more than one pair of terminals, and all terminals should be connected. Each grid is only available for one path. If there exists any grid with two path passing, like Figure 2, you may need to rip-up and re-route, like Figure 3.
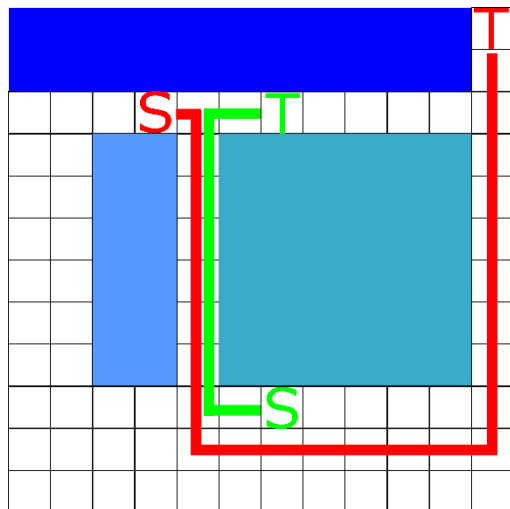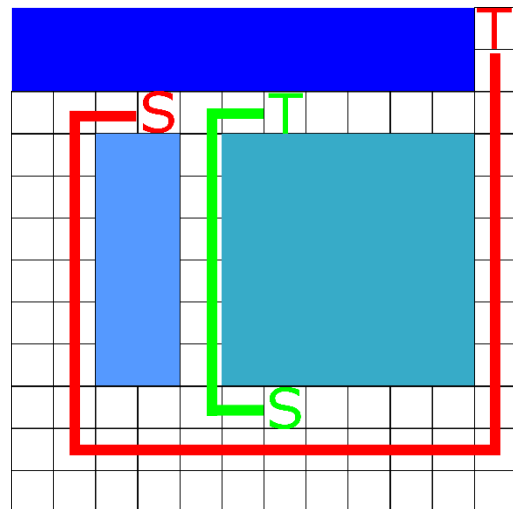

Figure 2

Figure 3

You should make sure all the terminals are connected, and minimize the wiring grids. The less grids you use, the higher score you get.

Input/Output Format Input:

Input file(*.in) is provided per case, and you should output the routing result to a output file(named by command).

We assume the lower-left corner of the table is the origin(0,0).

Take Figure 3 for example.

*.in:

```
1    .row 12              // Routing region
2    .col 12
3
4    .blk 3               // 3 blockages
5    2 3 3 8              // Left_down_X Left_down_Y Right_up_X Right_up_Y
6    5 3 10 8
7    0 10 10 11
8
9    .net 2               // 2 nets
10   Net1 3 9 11 11       // Source_X Source_Y Target_X Target_Y
11   Net2 6 2 6 9
```

*.out:

```
1    Net1
2    begin
3    29                   // Grid usage
4    3 9 1 9              // Source_X Source_Y Target_X Target_Y
5    1 9 1 1
6    1 1 11 1
7    11 1 11 11
8    end
9    Net2
10   begin
11   10
12   6 2 4 2
13   4 2 4 9
14   4 9 6 9
15   end
```

Environment:

1. Linux (Please make sure your code is available on our linux server. If it cannot be executed, you will get zero point!!)

2. Makefile and Readme should be provided

3. Naming rule:
   A. Name of the binary after "make" – studentID_lab2.o
   B. Execution procedure:
      ./studentID_lab2.o [input.in] [output file name]
   C. Not following specified naming rule will receive zero point

Ranking:

If you connect all the terminals correctly, you can get at least 60 points, and the fewer grids you use, the higher score you get.

You must write your own code. Copying codes may result in zero point.

Delay submission

Within 3 days: score * 0.7

More than 3 days: score = 0

Submission:

Please upload the following materials in a "zip" file to New E3 by the deadline, specifying your student ID in the subject field.

1. Source code (.cpp, .h)
2. Makefile
3. Executable binary
4. A Readme file (Information to how to make and execute your code)

Due date:

4/29

Any question:

Please do not hesitate to ask in the discussion forum(討論區).