# Lab 3 - Maze Router

Deadline: 2023/05/09

## *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 Fig.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.
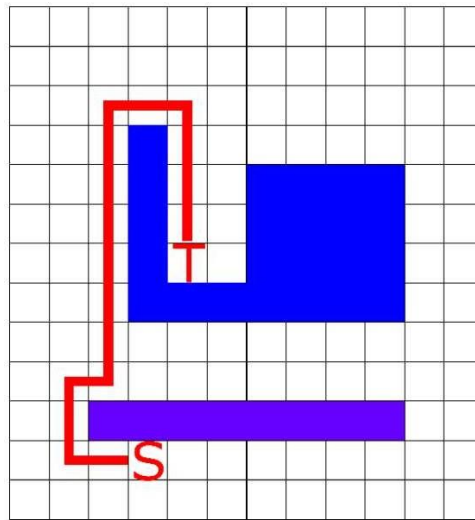


Fig. 1

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 Fig. 3.
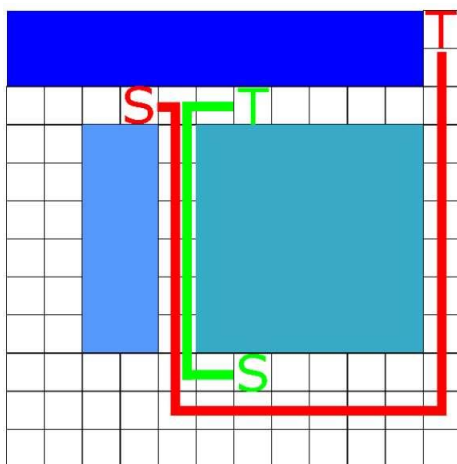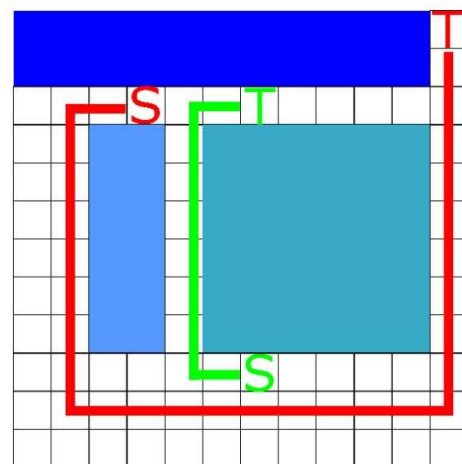


Fig. 2                                                    Fig. 3

You should make sure all the terminals are connected, and minimize the wiring grids.

## Input/ Output Format Input

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

We assume the lower-left corner of the grid is the origin (0,0). Take Fig. 3 as an example.

$5 <= row <= 100, 5 <= col <= 100, 1 <= block <= 50, 1 <= net <= 50$

Input

```
1   .row 12          // Grid size
2   .col 12
3
4   .block 3          // Number of block
5   2 3 3 8          // LeftDownX RightUpX LeftDownY RightUpY
6   5 10 3 8
7   0 10 10 11
8
9   .net 2
10  net1 3 9 11 11   //sourceX sourceY targetX target
11  net2 6 2 6 9
```

Output

```
1   net1 29       // Grid usage
2   begin
3   3 9 1 9       // sourceX sourceY targetX targetY
4   1 9 1 1
5   1 1 11 1
6   11 1 11 11
7   end
8   net2 10
9   begin
10  6 2 4 2
11  4 2 4 9
12  4 9 6 9
13  end
```

## Evaluation

1.  You MUST WRITE YOUR OWN CODE. Copying codes may result you to FAIL this course.
2.  Naming rule.
    A.  Name of the binary after "make" – lab3
    B.  Execution procedure: ./lab3 [input.in] [output file name]
    C.  Not following specified naming rule will receive zero mark
    D.  Hidden cases will be evaluated
3.  A drawing program is released to check your results.
    ./draw [input] [output]
    It will generate an SVG file; you can open it on the website.
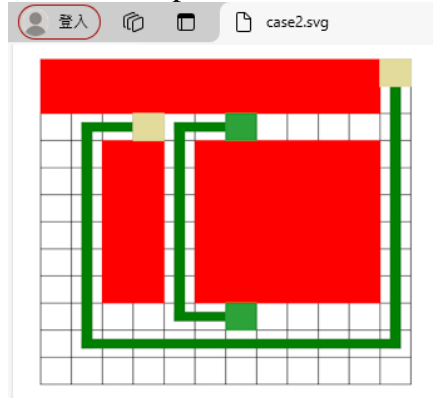
Please use "chmod 755 draw" first



The SVG file will be generated in the case2 folder.
The output of the drawing program:
Net, valid, grid usage in output, grid usage calculated by draw program

You should open the SVG file on the website to check for any crossings.



## *Program Submission*

Please upload the following materials in a .zip file (Student_ID.zip) to New E3 by the deadline, specifying your student ID in the subject field. Please ensure that your program can successfully execute on our server after compiling it. (If your submission file is not .zip file, you will get zero point!!)

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

## *Grading Policy*

1. For each case, the run time limit is up to 2 min. It will be regarded as "failed" if you use more than 2 min.
2. If you connect all the terminals correctly, you can get at least 60 points
3. The fewer grids you use, the higher score you get.

## *Notices*

1. Due Date: 2024/05/09 23:55:00
2. Please make sure your code is available on our Linux server. If it cannot be executed, you will get zero point.
3. Accept four days late submission, 10% deduction per day. Submission will not be accepted after 05/12.
4. Plagiarism is strictly forbidden, zero-point guarantee.