

Programming Assignment #4

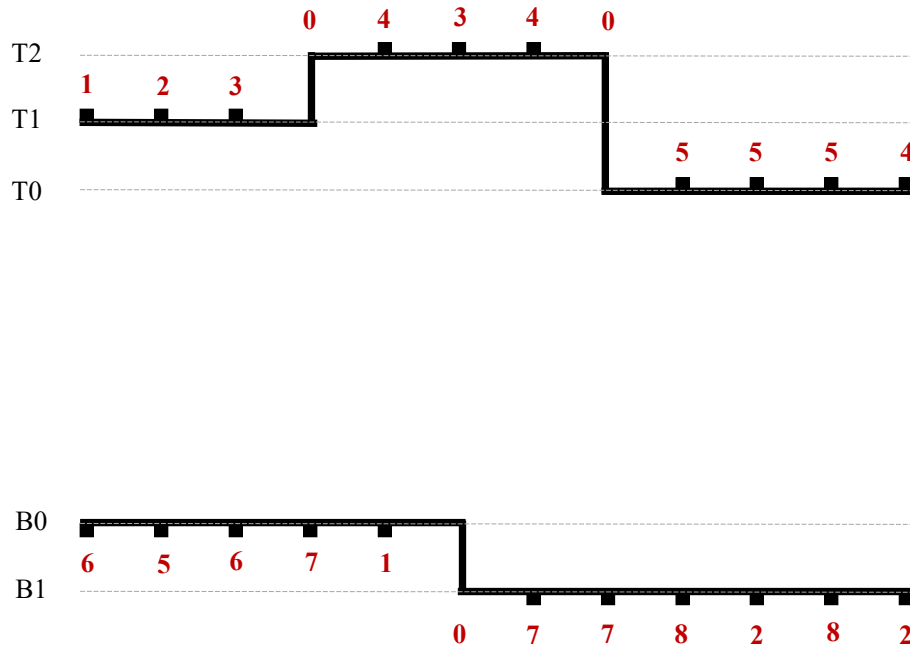
Channel Routing

1 Problem Description

In this programming assignment, you are asked to solve the 2-layer channel routing problem defined in the lecture notes. However, we assume that the top and bottom channel boundaries may not be straight, which could be rectilinear but irregular.

2 Input

The figure below gives an example channel routing instance.



The input format for the given channel routing instance is defined as follows:

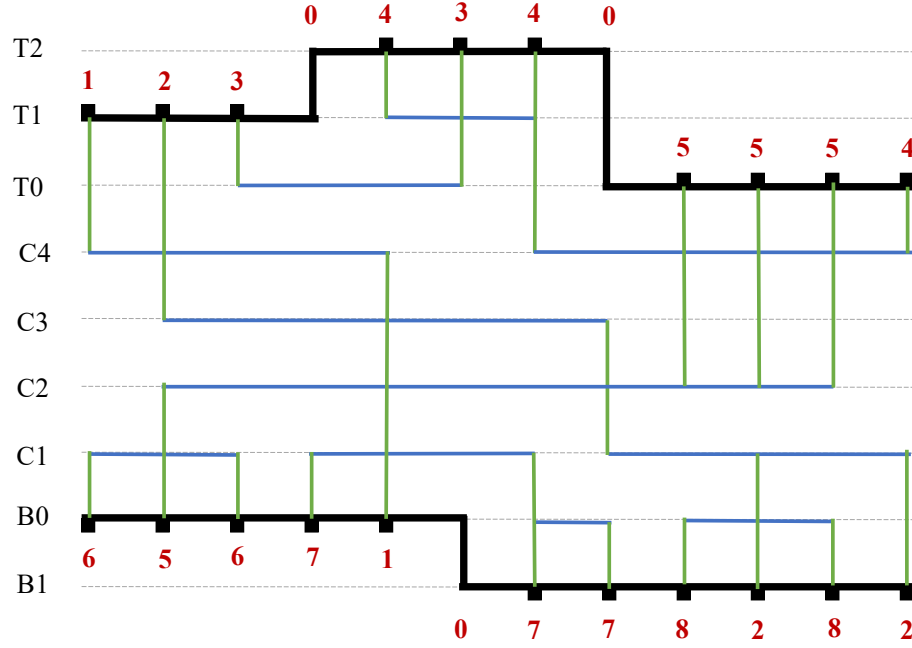
```

T1 0 3          // the 1st top boundary line at T1 from 0 to 3
T2 3 7          // the 2nd top boundary line at T2 from 3 to 7
T0 7 11         // the 3rd top boundary line at T0 from 7 to 11
B0 0 5          // the 1st bottom boundary line at B0 from 0 to 5
B1 5 11         // the 2nd bottom boundary line at B1 from 5 to 11
1 2 3 0 4 3 4 0 5 5 5 4 // the net IDs on the top boundary
6 5 6 7 1 0 7 7 8 2 8 2 // the net IDs on the bottom boundary

```

3 Output

The figure below gives an example routing result based on the given channel routing instance, where doglegs are allowed.



The output format for the routing result is defined as follows:

```

Channel density: 4 //The result requires 4 extra routing tracks in the channel.
Net 1
C4 0 4           // A horizontal wire at C4 from 0 to 4
Net 2
C3 1 7           // A horizontal wire at C3 from 1 to 7
C1 7 11          // A horizontal wire at C1 from 7 to 11
Dogleg 7         // A dogleg at 7 connecting both wires at C1 and C3
Net 3
T0 2 5           // A horizontal wire at T0 from 2 to 5
Net 4
T1 4 6           // A horizontal wire at T1 from 4 to 6
C4 6 11          // A horizontal wire at C4 from 6 to 11
Dogleg 6         // A dogleg at 6 connecting both wires at T1 and C4
Net 5
C2 1 10          // A horizontal wire at C2 from 1 to 10
Net 6
C1 0 2           // A horizontal wire at C1 from 0 to 2
Net 7
C1 3 6           // A horizontal wire at C1 from 3 to 6
B0 6 7           // A horizontal wire at B0 from 6 to 7
Dogleg 6         // A dogleg at 6 connecting both wires at C1 and B0
Net 8
B0 8 10          // A horizontal wire at B0 from 8 to 10

```

4 Language/Platform

1. Language: C or C++ is preferred.
2. Platform: Linux.

5 Command-line Parameter

In order to test your program, you are asked to add the following command-line parameters to your program :

```
./Lab4 [input file name] [output file name]
```

6 Submission

You need to submit a tar file, which includes (1) source codes, (2) Makefile, (3) a text readme file (readme.txt) stating how to build and use your program, to E3 (<https://e3.nycu.edu.tw/>) by the deadline. Please put all required files in a folder, and use the following command to compress the folder in the linux environment.

```
tar cvf Student_ID.tar Student_ID
```

The folder name and the compressed file name must be your student ID.

7 Grading Policy

This programming assignment will be graded based on (1) the **correctness**, (2) **solution quality**, and (3) **running time**. For each test case, it will be regarded as “failed” if your program takes more than 1 hours.