

Guidelines for Programming Assignment #3
(due on-line at 23:00, Dec. 2nd, 2020)

Submission: email to Prof. Chen, csj@ntu.edu.tw

Problem

Given a netlist and a 2-layer routing channel (in HV reserved model), you are asked to implement a **channel router to find the routing paths** for the nets and **minimize the number of used horizontal tracks**. You can implement constrained left-edge routing algorithm, dogleg routing algorithm, greedy routing algorithm, robustness routing algorithm or any other routing algorithm.

The following is an example where dogleg routing algorithm was applied. In the example, the number of used horizontal tracks is two and the number of failed nets is zero.

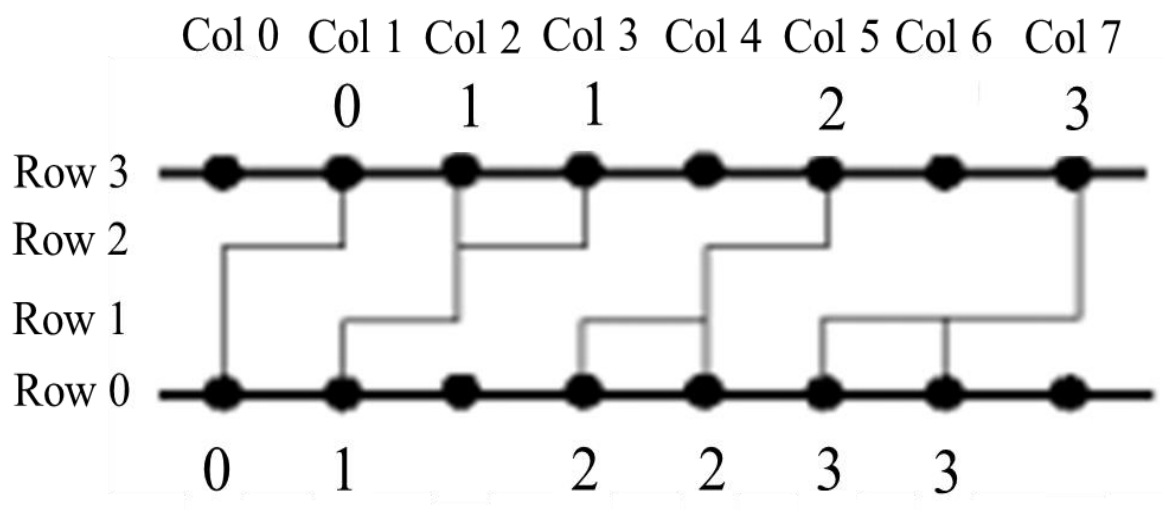


Figure 1

Input

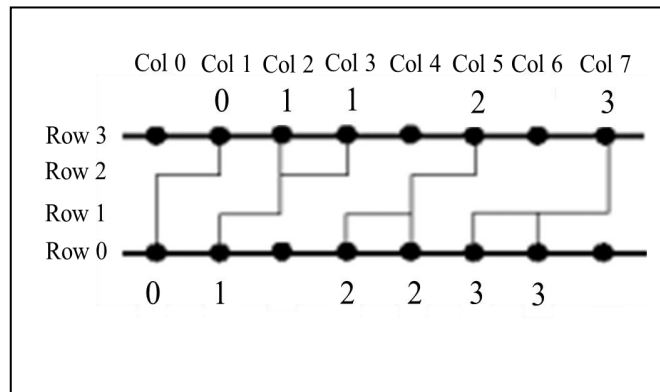
The first and second lines of the input file are the number of nets and the length of the channel. The description of each net contains the keyword NET, followed by the net name and a list of pins, and finally the symbol ‘;’. Note that we use a row# and a column# to describe the location of each pin. “b” represents a bottom row and “t” represents a top row. For example, “b3” represents that the pin is at the third column of the bottom row.

The input format is as follows:

```
NumNet [#net]
ChannelLength [channel length]
NET <Net Name> [<Pin# on bottom row>]* [<Pin# on top row>]*;
(* means one or more occurrences for a multi-pin net)
```

A sample input corresponding to Figure 1

```
NumNet 4
ChannelLength 8
NET n0 b0 t1;
NET n1 b1 t2 t3;
NET n2 b3 b4 t5;
NET n3 b5 b6 t7;
```



Output

In the program output, you are asked to give the number of used horizontal tracks and the routing path of each net. Please use the following format to describe the routing path:

- (1) Horizontal trunk: (Left end-column index, right end-column index, row index, H).
The row index is increasing from bottom to top. The bottom row is Row 0 and the row next to the bottom row is Row 1. “H” represents a horizontal trunk. For example, in Figure 1, the horizontal trunk of Net 0 is denoted as (0, 1, 2, H).
- (2) Vertical branch: (bottom end-row index, top end-row index, column index, V).
The column index is increasing from left to right. The leftmost column is Col 0. “V” represents a vertical branch. For example, in Figure 1, the vertical branches of Net 0 are denoted as (0, 2, 0, V)(2, 3, 1, V).

The first line of the output file is the number of used horizontal tracks. The description of the routing path of a net contains the keyword PATH, followed by the net name, the horizontal trunks and the vertical branches of the routing path.

The output format is as follows:

[Number of used horizontal tracks]
 PATH <Net Name> [Horizontal track]* [Vertical track]*;

A sample output file describes the routing result in Figure 1

2
 PATH n0 (0,1,2,H)(0,2,0,V)(2,3,1,V);
 PATH n1 (1,2,1,H)(2,3,2,H)(0,1,1,V)(1,3,2,V)(2,3,3,V);
 PATH n2 (3,4,1,H)(4,5,2,H)(0,1,3,V)(0,2,4,V)(2,3,5,V);
 PATH n3 (5,7,1,H)(0,1,5,V)(0,1,6,V)(1,3,7,V);

Required Files

You need to submit the following materials in a .tar.gz or .zip file:

- (1) Source codes and Makefile (you can divide the two algorithms into two different programs or integrate them into one program with command-line parameters)
- (2) Executable binary
- (3) A text readme file describing how to compile and run your programs
- (4) A report (**report.doc**) includes:

1. The number of used horizontal tracks, the number of failed nets, routability, CPU time, and memory used by your program using the following table:

	#horizontal tracks	#failed nets	Time (Sec)	Routability (%)	Memory (MB)
n4					
n10					
n30					
hidden case					

$$\text{Routability} = \frac{\# \text{routed nets}}{\# \text{total nets}}$$

2. The data structure and algorithms you used in your program.

Language/Platform

Language: C or C++. It is better that your program is executable on the EDA union Lab machine.

File-name Rule

The submission filename should be <student_id1>_<student_id2>-<p4>.tar.gz or <student_id1>_<student_id2>-<p4>.zip (e.g., **r02943000-p4.zip**). If you have a modified version, please add -v[version_number] as a postfix to the filename and resubmit (e.g., **r02943180_r02943000-p4-v1.zip**).

Command-line Parameter

You have to add command-line parameters in your program to specify the input and output file names as the following format:

[executable_file_name] [input_file_name] [output_file_name]

(e.g., ChannelRouter input.dat output.dat)

Evaluation

The individual score per test case is determined by the following items:

- (1) routability,
- (2) number of used horizontal tracks,
- (3) run time,
- (4) output file format,
- (5) report.doc, and
- (6) demo in class.

If there is any question, please mail to csj@ntu.edu.tw