

# Lab2 –Partition

**Deadline: 2024/4/15**

## Lab 2 Introduction

This lab is an exercise lab to implement partition algorithm, and you have to compare your result with hMetis. Using K-L, F-M, SA, or any partitioning algorithm you know.

## Input

Given a netlist input file

### **Example (input.txt)**

```
3 5
1 2 3
2 4
1 5
```

## Explanation of the Input

1. The first line represents how many nets, how many nodes there are  
(Ex. 3 means 3 nets, 5 means 5 nodes)
2. The next line to the last line defines the nodes to which the net is connected.  
(Ex. 1 2 3 means the first net connect to node 1, 2 and 3.)

## Output

### **Output Format (output.txt)**

```
1
0
1
0
1
```

## Explanation of the Output

1. The first line to the last line defines the groups to which the node belongs.  
(Ex. The first line 1 means the first node (node 1) belongs to the group 1. The second line 0 means the second node (node 2) belongs to the group 0.

## Algorithm

You will need to partition the given nodes to two groups and try to **minimize the number of cut between the two groups while balancing number of nodes**. The balance factor is  $0.45 \sim 0.55$ .

### Evaluation

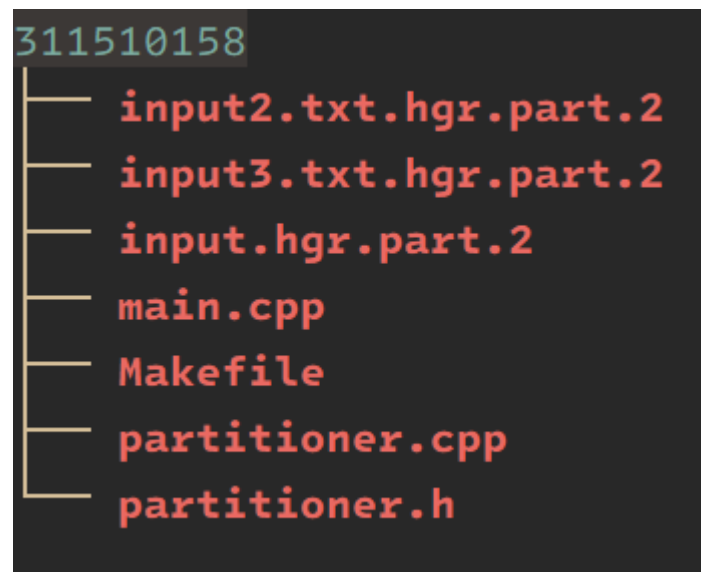
1. You **MUST WRITE YOUR OWN CODE**. Copying codes may result you to FAIL this course.
2. Naming rule.
  - A. Name of the binary after “make” – Lab2
  - B. Execution procedure (**on the server**):
    - i. make
    - ii. ./Lab2 [input]
  - C. Name of the output file – **output.txt**
  - D. **Not following specified naming rule will receive zero mark**
3. Hidden cases will be evaluated
4. A verifier is released to evaluate your result.  
./verifier [input] [output]  
(Please make sure that your output results can pass the verifier)

### Program Submission (If your submission file is not following the rules, you will get zero point!!)

Please put the following materials in a folder, which name should be your student ID, then compress **the whole folder** in a .zip file, which name should also be your student ID. Finally, upload the zip file to New E3 by the deadline

1. Source code (\*.cpp, \*.h).
2. Makefile
3. The result of each test case generated from hMetis
  1. Use *./shmetis [input] [ # of groups ] [ balance factor ]* to generate the result, [# of groups] is 2 and [balance factor] is 5 in this homework.
  2. The file name is same as the output of hMetis, namely "input.hgr.part.2", "input2.txt.hgr.part.2" and "input3.txt.hgr.part.2" for "input.hgr", "input2.txt.hgr" and "input3.txt.hgr", respectively

Example:



### Grading Policy

We will determine your score according to the minimum cut result and the run time for 4 test case. (Priority: minimum cut result > run time (7:3 after normalizing each factor))

1. For each case, the run time limit is up to 30 seconds. It will be regarded as “failed” if you use more than 30 seconds.
2. If you can generate a legal solution, you can get at least 70 at that case.
3. Random output is forbidden.
3. Result generated from hMetis or other external partitioners is forbidden.

### Notices

- Due Date : 2024/04/15 23:55:00
- Please make sure your code is executable on our linux server. If it cannot be executed, you will get zero point.
- Accept three days late submission, 10% deduction per day.  
Submission will not be accepted after 4/18 23:55:00.
- Plagiarism is strictly forbidden. 0 grade guarantee.