Introduction to Algorithms
알고리즘개론
2018 Spring Semester

Jinkyu Lee

Dept. of Computer Science and Engineering (and Software),
Sungkyunkwan University (SKKU)

Rules for all homework

- You should follow instructions.
  - Complier
    - You will get no point if your program cannot be complied with the specified complier
  - Input/output format
    - You will get no point if TA's automatic evaluation program cannot parse your input or output.
  - Permitted modification scope
    - You will get no point if you modify code outside of the permitted modification scope
  - All other rules
    - You will get severe penalty or no point if you violate the given rules.

Complier and input/output rules for all homework

- Every implementation homework will be evaluated by TA's automatic evaluation program with the following complier.
  - Complier: GCC 6.3
  - You will get no point if your program cannot be complied with GCC 6.3.
  - You can use standard library such as *stdlib.h* and *math.h*.
- Input/output format
  - You will get no point if TA's automatic evaluation program cannot parse your input or output according to the following rules.
  - Use stdin and stdout
  -
    -
  -
    -
- Recommended development environment (Windows)
  - IDE: CodeBlocks (http://www.codeblocks.org/downloads/26)
  - Compiler: MinGW (https://sourceforge.net/projects/mingw)
  - You can use the corresponding compliers for Linux and Mac.

Homework 4

- 7.5 points (7.5%)
  - 4A: 1.5 points (1.5%)
  - 4B: 3.5 points (3.5%)
  - 4C: 2.5 points (2.5%)
- Due data: 2018/6/11 Monday 23:59
  - Delay penalty: 1% per hour
  - Delay and evaluation will be applied to each file.
  - TA will only evaluate the latest version of your homework with time stamp.
  - Your time management is very important!
- Submission to iCampus
- TA: Jaeheon Kwak
  - 0jaehunny0@gmail.com

Homework 4

- 4A
  - No file submission
- 4B
  - Code: Yourid_HW4B.c
    - The file type should be c, not cpp.
    - The file should be a single file.
    - Submit to “Homework 4B – Code”
  - Report: Yourid_HW4B.hwp
    - The file type can be hwp, doc(x) or pdf, not others
    - Submit to “Homework 4B – Report”
- 4C
  - Code: Yourid_HW4C.c
    - The file type should be c, not cpp.
    - The file should be a single file.
    - Submit to “Homework 4C – Code”
  - Report: Yourid_HW4C.hwp
    - The file type can be hwp, doc(x) or pdf, not others
    - Submit to “Homework 4C – Report”

Homework 4A

- 1.5 points (1.5%)

- You will have a in-class quiz in 5/28 (Mon), 5/30 (Wed) or 6/4 (Mon).
  - The coverage is all contents in Lecture Note 12 and 13.
  - If you have any reasonable possibility to be absent in those days, please tell me as soon as possible.
  - You will get no point if you miss the quiz.

Homework 4B

- Solve a Problem: Animal Ranking List
  - David is a faithless zookeeper. He wastes his time comparing which animals are stronger, at work time. To prevent him from wasting time anymore, you decide to make an animal ranking list.

  - You designed some rules of the animal ranking list.
    - Animal's ranking is determined by several comparisons that inform which one of the two animals is stronger than another.
    - If an animal **A1** is stronger than **A2**, and **A2** is stronger than **A3**, then **A1** is stronger than **A3** (If **A1** $>$ **A2**, and **A2** $>$ **A3**, then **A1** $>$ **A3**).
    - If an animal is not weaker than any other animals, its ranking is **1**.
    - If there are multiple possible rankings, choose the smallest number.
    - You cannot make an animal ranking list when any cycle exists in the comparisons.
    - There may be several animals with the same ranking.
    - The number of ranking is always continuous.

  - You are given **N** animals and **C** comparisons, make an animal ranking list.

Homework 4B

- Input
  - The input starts with two integers **N** and **C**, the number of animals and the number of comparisons between them.
  - The second line of the input is a string, which includes animals' name.
  - The next **C** lines consist of two animal name **A1** and **A2**, meaning that the animal **A1** is stronger than the animal **A2**.
  - Animal names never ends with white space.
  - The names are separated by slash (/).
  - Animal names are never duplicated.
  - Example

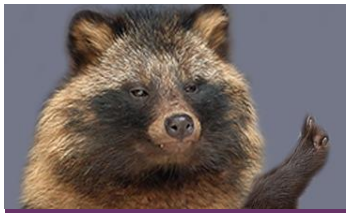
    3 2

    Raccoon dog/Raccoon/Red Panda

    Raccoon/Red Panda

    Raccoon/Raccoon dog

Raccoon



Raccoon dog

Red Panda

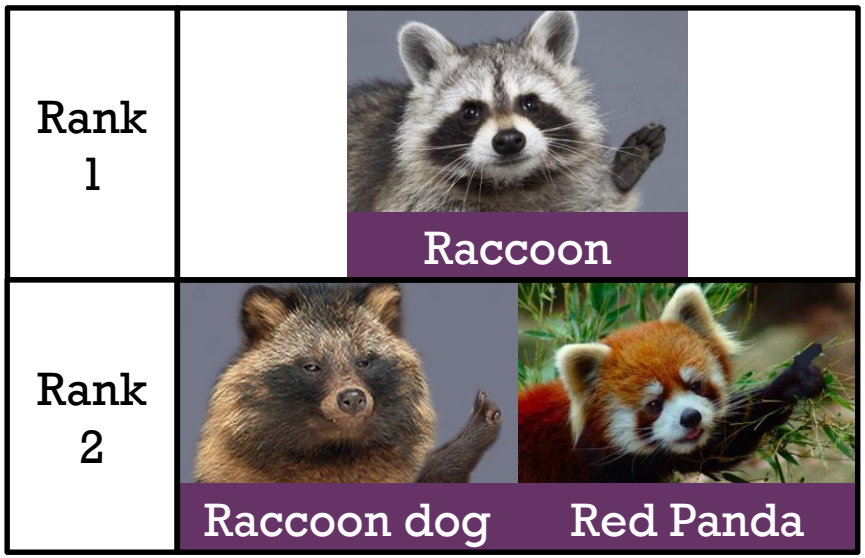
Homework 4B

- Output
  - The output should contain **N** lines. Each line contains the ranking of the animal, a space, and the name of the animal, from the smallest number to the largest number. If there is the same ranking for several animals, then sort them in a lexicographical order.
  - When you cannot make the animal ranking list, just print “Stupid David!\n”.
  - Example

    1 Raccoon

    2 Raccoon dog

    2 Red Panda



- Constraints
  - $1 \leq N < 100$
  - $1 \leq C < 1000$
  - $1 \leq length\ of\ animal\ name < 20$

Homework 4B

- Sample input & output 1
  - Input

5 4

Raccoon dog/Raccoon/Red Panda/Dog/Panda

Raccoon/Red Panda

Raccoon/Raccoon dog

Red Panda/Panda

Raccoon dog/Dog

  - Output

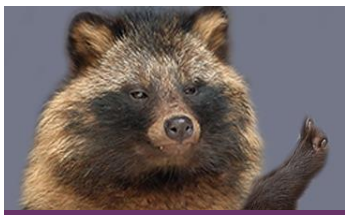
1 Raccoon

2 Raccoon dog

2 Red Panda

3 Dog

3 Panda



Jinkyu Lee
Dept. of Computer Science and Engineering

Homework 4B

- Sample input & output 2
  - Input

    3 3

    Raccoon dog/Raccoon/Red Panda

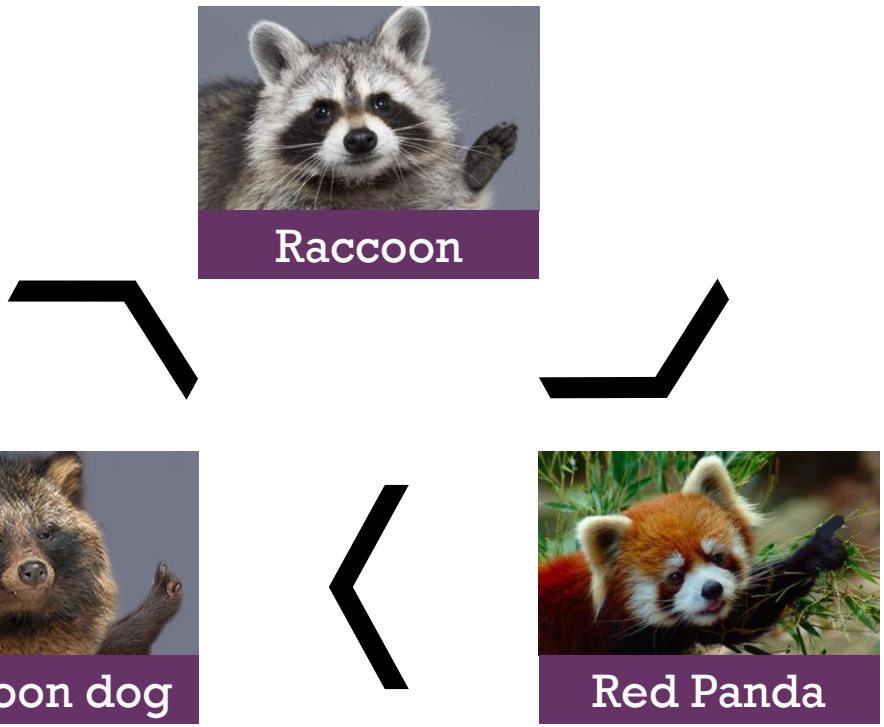
    Red Panda/Raccoon dog

    Raccoon/Red Panda

    Raccoon dog/Raccoon

  - Output

    Stupid David!



Homework 4B

- Sample input & output 3
  - Input

5 4

Raccoon dog/Raccoon/Red Panda/Dog/Panda

Raccoon/Raccoon dog

Red Panda/Panda

Raccoon dog/Dog

Dog/Panda

  - Output

1 Raccoon

1 Red Panda

2 Raccoon dog

3 Dog

4 Panda

Homework 4B

- Total score: 3.5 points (3.5%)
- Performance evaluation (3.0 points)
  - TA will test several cases.
  - For each case, the result should be printed within 10 seconds.
  - Your C code is tested with the following complier.
    - GCC 6.3
    - You will get zero point if your program cannot be complied with GCC 6.3.
  - You should follow the input and output format.
    - You will get zero point if the TA's automatic evaluation program cannot parse your input or output.

Homework 4B

- Report evaluation (0.4 points)
  - Explain your code using an example
  - No more than 2 pages
  - In English or Korean
- Code readability (and rules) evaluation (0.1 points)
  - Indent properly
  - Use meaningful names of variables
  - Write sufficient comments in English
    - Do not include any other natural language than English in you code.
  - Use correct file names

Homework 4C

- Implement Prim's algorithm as follows:
  - You will be given 4 files
    - "main. c"
    - "header. h"
    - "implemented. c"
    - "prim. c" (Blank File)
  - Implement 4 functions in "prim. c"
    - void minHeapify(MinHeap * M, int Index)
    - Vertex * extractMin(MinHeap * M)
    - void decreaseKey(MinHeap * M, int V, int Key)
    - void addVertexToHeap(MinHeap * M, Graph * G, int V)
    - int primAlgorithm(Graph * G)

Homework 4C

- There are 7 functions are implemented in "implemented. c"
  - Vertex * initVertex(Vertex * Vert, int N, int V)
  - Graph * initGraph(int V)
  - void connectVertexes(Graph * G, int ParaA, int ParaB, int ParaW)
  - MinHeap * initMinHeap(int V)
  - void swapVertex(Vertex ** ParaA, Vertex ** ParaB)
  - int isEmptyHeap(MinHeap * M)
  - int isInMinHeap(MinHeap * M, int V)

- How can I compile them? (example: Linux)
  - $ gcc main. c implemented. c prim. c – o out

- Replace "prim. c" filename with your student ID and submit it.
  - That is, Yourid_HW4C.c
  - Do not submit "main. c", "header. h" or "implemented. c".

Homework 4C

- Input
  - The first line contains two integers **V** and **E**, the number of vertexes and the number of edges. The next **E** lines consists of three integers, vertex **V1**, vertex **V2** and weight **W**. It means that vertex **V1** and **V2** is connected with weight **W**.
  - Example

    2 1

    0 1 1

- Output
  - The output will contain a single number representing the sum of the minimum spanning tree's edge weights.
  - Example

    1

- Constraints
  - $\mathbf{2 \le V \le 100}$, $\mathbf{1 \le E}$, $\mathbf{Output \le INT\_MAX}$

Homework 4C

- Sample input & output 1
  - Input

3 3

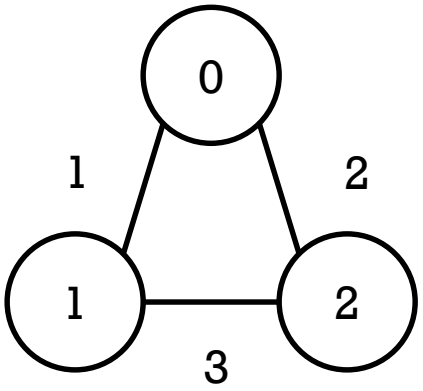
0 1 1

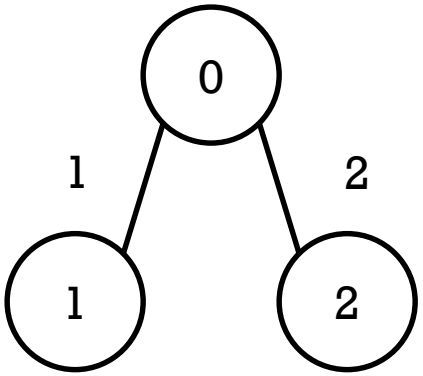
0 2 2

1 2 3

  - Output

3





Homework 4C

- Sample input & output 2
  - Input

5 6

0 1 3

0 2 8

1 2 8

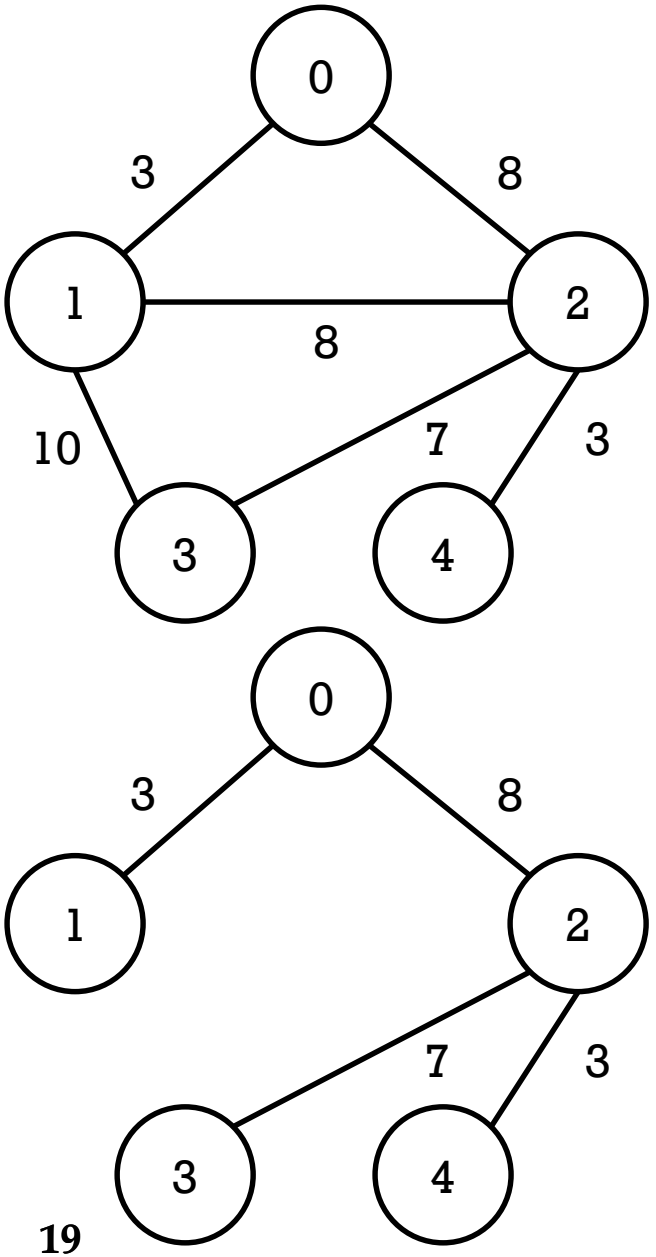
1 3 10

2 3 7

2 4 3

  - Output

21



Jinkyu Lee
Dept. of Computer Science and Engineering

Homework 4C

- Total score: 2.5 points (2.5%)
- Performance evaluation (2.1 points)
  - TA will test several cases.
  - For each case, the result should be printed within 10 seconds.
  - Your C code is tested with the following complier.
    - GCC 6.3
    - You will get zero point if your program cannot be complied with GCC 6.3.
  - You should follow the input and output format.
    - You will get zero point if the TA's automatic evaluation program cannot parse your input or output.

Homework 4C

- Report evaluation (0.3 points)
  - Explain your code using an example
  - No more than 2 pages
  - In English or Korean
- Code readability (and rules) evaluation (0.1 points)
  - Indent properly
  - Use meaningful names of variables
  - Write sufficient comments in English
    - Do not include any other natural language than English in you code.
  - Use correct file names