CS2040C AY18/19 Semester 1 Practical Exam

3rd November 2018
1300h to 1500h

Instructions

- The exam consists a total of **2 tasks** (with subtasks). Attempt both tasks.
- You can use any editor on the Lab computers.
- You can refer to C++ documentations which are found on the Desktop.
- For each task/subtask, you are to submit your working solutions to **Mooshak**.
    - **URL** for Mooshak: http://cs2040c.comp.nus.edu.sg/~mooshak/
    - **Username**: Matric Number in **UPPERCASE (Eg: A0….X)**
    - **Password**: First 4 letters of your usual Mooshak password + Last 4 letters of your NUSNET ID. If your Mooshak Password is "**Ab1D**3xyz" and your NUSNET ID is "E0179634", then your password is "**Ab1D**9634".
    - Raise your hand if you need assistance logging into Mooshak.
- You may make any number of attempts/submissions. Only the last submission per subtask will be assessed.
- The total amount of marks for this Practical Exam is 12 marks.

Task Overview

Task 1 – Frequencies (5m)

- A (Medium, 3m): Full task that also solves B
- B (Easy, 2m): Easier variant (does not solve A).

It is **not possible** to obtain partial score for 'A' unless 'B' is awarded full score.
If you **cannot** solve 'A', you are advised to attempt 'B' first.

Task 2 – Class Photo (7m)

- C (Medium, 3m): Medium variant (does not solve D and E).
- D (Easy, 2m): Easier variant (does not solve C and E).
- E (Hard, 2m): Full variant that also solves C and D.

It is **not possible** to obtain partial score for 'E' unless both 'C' and 'D' are awarded full score.
If you **cannot** solve 'E', you are advised to attempt 'C' and 'D' first.
You **can** be awarded scores for 'C' and 'D' independently.

Task 1 (A): Frequencies

Time Limit: 1s
Memory Limit: 512MB

Input Description

In the first line, you are given an integer **N** ($1 \leqslant$ **N** $\leqslant 100\,000$).

In the second line, you are given **N** positive integers (each integer in the second line is between 1 and 2 Billion, i.e. $2*10^9$ and **not necessarily distinct**)

Output Description

Based on the given input, print one line for every distinct integer in the input. The line should contain 2 integers: the frequency of the integer followed by the value of the integer separated by a space.

Print these lines in decreasing order of frequency.

If more than one integer has the same frequency, print them in increasing order of value.

Sample Input

```
10
4 2 2 40 2 4 4 20 2 1
```

Sample Output

```
4 2
3 4
1 1
1 20
1 40
```

Explanation

'2' appears 4 times, '4' appears 3 times and '1', '20', '40' appear once.

Task 1 (B): Frequencies

Simpler variant of A

Time Limit: 1s

Memory Limit: 512MB

The Delta

Everything is the same as part A. However, in part B, the **N** positive integers are **guaranteed to be distinct**.

Sample Input

```
3
4 2 5
```

Sample Output

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

Task 2 (C): Class Photo

Time Limit: 1s
Memory Limit: 512MB

Prologue

CS2040C is coming to a close and Rar the Cat wants to take a class photo with all the students.
To make the photo nice, he wants to arrange the students in increasing height from left to right.
Rar the Cat stands at the **leftmost position** of the photo and since he is a cat, he is very short and **effectively has height 0**.
He sees the students arrive one by one and orders each of them to stand **to the right** of the correct person.

Input and Output Description

In the first line, you are given an integer **N** ($1 \leqslant \mathbf{N} \leqslant 100\,000$).
The next **N** lines each contains a string and an integer representing the **name** and the **height** of each student in the order they enter the classroom.
For each student, print the name of the student they should stand directly to the right of. You are to ensure that the students are always ordered in increasing height from left to right.
Once all **N** students have arrived, print the names of the students in the photo from **left to right** separated by spaces on a single line. **Do NOT print a space at the end of this line.**

All names in the input are given in UPPERCASE ['A'..'Z'] and short (not more than 10 characters each) and are distinct. All heights in the input are between 1 and 2 Billion, i.e. $2*10^9$ and are distinct.

Sample Input

```
5
GARY 3
STEVEN 6
PANDA 4
SHARK 10
TURTLE 1
```

Sample Output

```
RAR
GARY
GARY
STEVEN
RAR
RAR TURTLE GARY PANDA STEVEN SHARK
```

Explanation

1. Initially, the classroom only has Rar the Cat with height 0:
   [RAR, 0]
2. After Gary enters the classroom, he should stand on the right of RAR since Gary is taller:
   [RAR, 0], [GARY, 3]
3. After Steven enters the classroom, he should stand on the right of GARY since Steven is taller:
   [RAR, 0], [GARY, 3], [STEVEN, 6]
4. Panda's height of 4 is in between Gary and Steven. Hence, he should stand on the right of GARY:
   [RAR, 0], [GARY, 3], [PANDA, 4], [STEVEN, 6]
5. Shark should stand on the right of STEVEN since Shark is taller:
   [RAR, 0], [GARY, 3], [PANDA, 4], [STEVEN, 6], [SHARK, 10]
6. Turtle's height of 1 is in between Rar and Gary. Hence, should stand on the right of RAR:
   [RAR, 0], [TURTLE, 1], [GARY, 3], [PANDA, 4], [STEVEN, 6], [SHARK, 10]

Task 2 (D): Class Photo

Modified variant of C

Time Limit: 1s

Memory Limit: 512MB

The Delta

The students' heights are **all the same**.

If there are multiple possible positions, the student will stand at the position **as close to Rar the Cat as possible** (i.e. choose the leftmost possible position).

Sample Input (Same height)

```
2
GARY 3
STEVEN 3
```

Sample Output

```
RAR
RAR
RAR STEVEN GARY
```

Explanation

1. Before anybody enters the class, the line looks like this:
   [RAR, 0]
2. After Gary joins the queue:
   [RAR, 0], [GARY, 3]
3. After Steven joins the queue:
   [RAR, 0], [STEVEN, 3], [GARY, 3]

Task 2 (E): Class Photo

Combination of C and D

Time Limit: 1s

Memory Limit: 512MB

The Delta

The students' heights are **not necessarily distinct**.

If there are multiple possible positions, the student will stand at the position **as close to Rar the Cat as possible** (i.e. choose the leftmost possible position).

Sample Input (Not necessarily distinct height)

```
5
GARY 3
STEVEN 6
PANDA 6
SHARK 10
TURTLE 3
```

Sample Output

```
RAR
GARY
GARY
STEVEN
RAR
RAR TURTLE GARY PANDA STEVEN SHARK
```

Explanation

1. Initially, the classroom only has Rar the Cat with height 0:
   [RAR, 0]
2. After Gary enters the classroom, he should stand on the right of RAR since Gary is taller:
   [RAR, 0], [GARY, 3]
3. After Steven enters the classroom, he should stand on the right of GARY since Steven is taller:
   [RAR, 0], [GARY, 3], [STEVEN, 6]
4. Panda's height of 6 is higher than Gary but same as Steven. He can either stand directly to the right of Gary or directly to the right of Steven to keep the line in increasing order. However, since standing on the right of Gary is closer to Rar the Cat than standing on the right of Steven, you should ask Panda to stand on the right of Gary instead:
   [RAR, 0], [GARY, 3], [PANDA, 6], [STEVEN, 6]
5. Shark should stand on the right of STEVEN since Shark is taller:
   [RAR, 0], [GARY, 3], [PANDA, 6], [STEVEN, 6], [SHARK, 10]
6. Turtle's height of 3 is the same as Gary. He can either stand directly to the right of Rar the Cat or directly to the right of Gary. Since standing on the right of Rar the Cat is closer to Rar the Cat than standing on the right of Gary, he should be asked to stand on the right of RAR instead:
   [RAR, 0], [TURTLE, 3], [GARY, 3], [PANDA, 6], [STEVEN, 6], [SHARK, 10]