

Programming assignment #2

寶可夢戰鬥力排行榜

Objective

1. To understand how to design an algorithm.
2. To exercise the concept of recursion.
3. To learn how to use red-black tree and augmenting data structures.

Problem

Adar 老師需要一個寶可夢即時戰鬥力排行榜，戰鬥力數值越高的排行越前面，戰鬥力數值相同的排名相同（排名=戰鬥力數值大於自己的個數+1）。此排行榜共有 6 個功能：

1. “I”: 將特定 ID 與戰鬥力的寶可夢輸入到當前排行榜中。
2. “D”: 從當前排行榜中移除特定 ID 與戰鬥力的寶可夢。
3. “R”: 輸出當前排行榜中特定排名的 ID 與戰鬥力。
4. “V”: 輸出當前排行榜中特定戰鬥力數值的排名。
5. “B”: 輸出當前排行榜中特定戰鬥力數值範圍的最高/最低排名。
6. “A”: 輸出當前排行榜中特定排名範圍內最排名高/最低的 ID 與戰鬥力。

Input

排行榜中的各指令如下表所示：

I <ID> <value>	// 將 ID 為<ID>、戰鬥力為<value>的寶可夢輸入 // 到排行榜中（不會輸入已存在排行榜中的 ID）
D <ID> <value>	// 將 ID 為<ID>、戰鬥力為<value>的寶可夢由排 // 行榜中移除（此寶可夢必存在排行榜中）
R <rank>	// 輸出排名為<rank>的 ID/戰鬥力（排名必存在）
V <value>	// 輸出戰鬥力為<value>的排名（數值必存在）
B <value 1> <value 2>	// 輸出戰鬥力在<value 1>到<value 2>之間的最高 // 與最低排名（包含<value 1>和<value 2>）
A <rank 1> <rank 2>	// 輸出排名在<rank 1>到<rank 2>之間排名最高 // 與最低的 ID/戰鬥力（包含<rank 1>和<rank 2>）

在 input file 中，一行代表一個指令，指令由前至後依序執行。

Input file example: (“/” is comment and will not appear in input file)

```
I 5 17          // 將 ID 為 1 戰鬥力為 17 的寶可夢輸入到排行榜中
I 2 8
I 16 12
R 2             // 輸出排名第 2 的 ID 與戰鬥力
I 4 4
I 11 4
D 16 12        // 將 ID 為 16 戰鬥力為 12 的寶可夢由排行榜中移除
B 3 16         // 輸出戰鬥力在 3 到 16 之間的最高/最低排名
I 6 15
V 8            // 輸出戰鬥力為 8 的排名
I 16 20
I 9 13
I 10 15
A 4 2          // 輸出排名在 4 到 2 之間排名最高/最低的 ID 與戰鬥力
```

Output

當需要輸出時，在最前方先輸出指令的代號，然後才進行輸出；若指令為“R”，先輸出 ID (遇到相同排名的寶可夢時則輸出其中 ID 最小者)，再輸出戰鬥力；若指令為“B”，先輸出最高排名，再輸出最低排名；若指令為“A”，先輸出排名最高的 ID 與戰鬥力，再輸出排名最低的 ID 與戰鬥力 (遇到相同排名的寶可夢時則輸出其中 ID 最小者)；若指令“B”或“A”所給的區間內找不到答案，則輸出一個 0 即可 (e.g., B 0, A 0)。

Output file example: (“/” is comment and will not appear in output file)

```
R 16 12        // 輸出排名第 2 的 ID 與戰鬥力
B 2 3          // 輸出戰鬥力在 3 到 16 之間的最高/最低排名
V 3            // 輸出戰鬥力為 8 的排名
A 5 17 6 15    // 輸出排名在 4 到 2 之間排名最高/最低的 ID 與戰鬥力
```

Compile & Execute

Compile command: **\$ g++ <hw2.cpp> -O3**

Execute command: **\$./<execute file> <input file> <output file>**

(Ex : \$./a.out input1.txt output1.txt)

Note that input and output file should be the arguments of program. Please make sure your code can be compiled and executed.

Program Submission

1. Please use **C++** language and your program **must be written in only one source file**.
2. Your source file must be named as “**Student_ID_number_hw2.cpp**” and please make sure that all characters of the filename are in **lower case**. For example, if your student number is 0710100, the name of your program file should be “**0710100_hw2.cpp**”.
3. **Do not print** any words on terminal.
4. Upload your **report** and **program** to the new E3 by deadline. (不要壓縮)
未按照繳交格式繳交作業的人會得到 0 分!!!

Report

1. No more than 3 pages.
2. Your report must contain:
 - a. The flow chart or the pseudo code of you program.
 - b. The experimental results and analysis. (Time complexity)
3. The report file name must be “**Student_ID_number_hw2.pdf**” and please make sure that all characters of the filename are in **lower case**. For example, if your student number is 0710100, the name of your program file should be “**0710100_hw2.pdf**”.

Grading

You need to submit both your source code and report. Remember the submission rules mentioned above, or **you will get zero point!!!!**

- Mode I-D-R (×3) 15%
- Mode I-D-V (×3) 15%
- Mode I-D-B (×3) 15%
- Mode I-D-A (×3) 15%
- *** Time limit: 1 minute for above cases**
- Big case (1 個) 20%
 - Correct answer 10%
 - Timing performance (If the answer is correct) 10%
 - * Time limit: 5 minute for big case**
- Report 20%

Due Date

- **Due Date : 2020/11/13 23:55:00**
- **Plagiarism is forbidden.**
- 遲交，第一天分數打 9 折 第二、三天打 8 折 第四、五天打 7 折 第六天打 6 折，之後則不予繳交

