Algorithm Analysis Homework 1

Due by 3/17(Fri.) through LMS

* Grading policy

Remember that correctness is an important criterion, but by no means the whole story. Grades on program will be based on:

1. Correct behavior on typical input: 70%
2. Adherence to specification: 10%
3. Correct behavior on extreme or unusual situations & reasonable recovery from unusual or incorrect inputs.: 10%
4. Readability ~ comments, Mnemonics identifier, clear program structure.: 10%

* Note

1. At header part of comment, list all the references you used when you do this homework.

   For ex)

   (1) 강의 slide chapter 6. page3-5

   (2) Blog: ** URL here **

   (3) book: “Algorithm analysis in C++” by Someone, chapter 5

2. You should not use ‘STL’.
3. No AI service

Write a program that implements min-priority queue. (You should use heap data structure, otherwise you will no point for this homework. Also element index should begin from 1, not 0.)

1) Assume maximum heap size is 30.
2) Each element has three fields – name, id, and school.
3) ‘name’ consists of English alphabet (at most 10 characters). You don’t need to check validity of ‘name’ (i.e. we assume user always enter alphabet letters whose length is less than or equal to 10. No extreme/unusal input.).
4) id is 4 digit (integer) number and used as a key field.

   case1) We assume user always enter integer, but might enter a number other than 4 digit number (such as 12345. If 1000’s number is 0, it is not 4 digit number, i.e. 0123 is not valid one.).

   case2) User may enter any character, such as “12 34”, “123o”, “12_34”, etc.

   If you choose to do your program with case2, you will get extra 20% points.
5) ‘school’ is either “Handong”, “Doodong”, or “Sedong”.

Your program should consists of following functions:

- INSERT(*S*, *x*) : inserts element *x* into set *S*.
- MINIMUM(*S*) : returns element of S with largest key.
- EXTRACT-MIN(*S*) : removes and returns element of S with largest key.
- DECREASE-KEY(*S*, *x*, *k*) : decreases value of element *x*'s key to *k*.

You may use other functions describe other than above functions.

When your program starts, it repeatedly presents menu until user enters 'Q'.
Following is description of each menu.

I: When this menu is chosen, program asks user's name and score, then inserts element into queue.

D: When this menu is chosen, program removes and displays record of element with smallest score and rebuild queue.

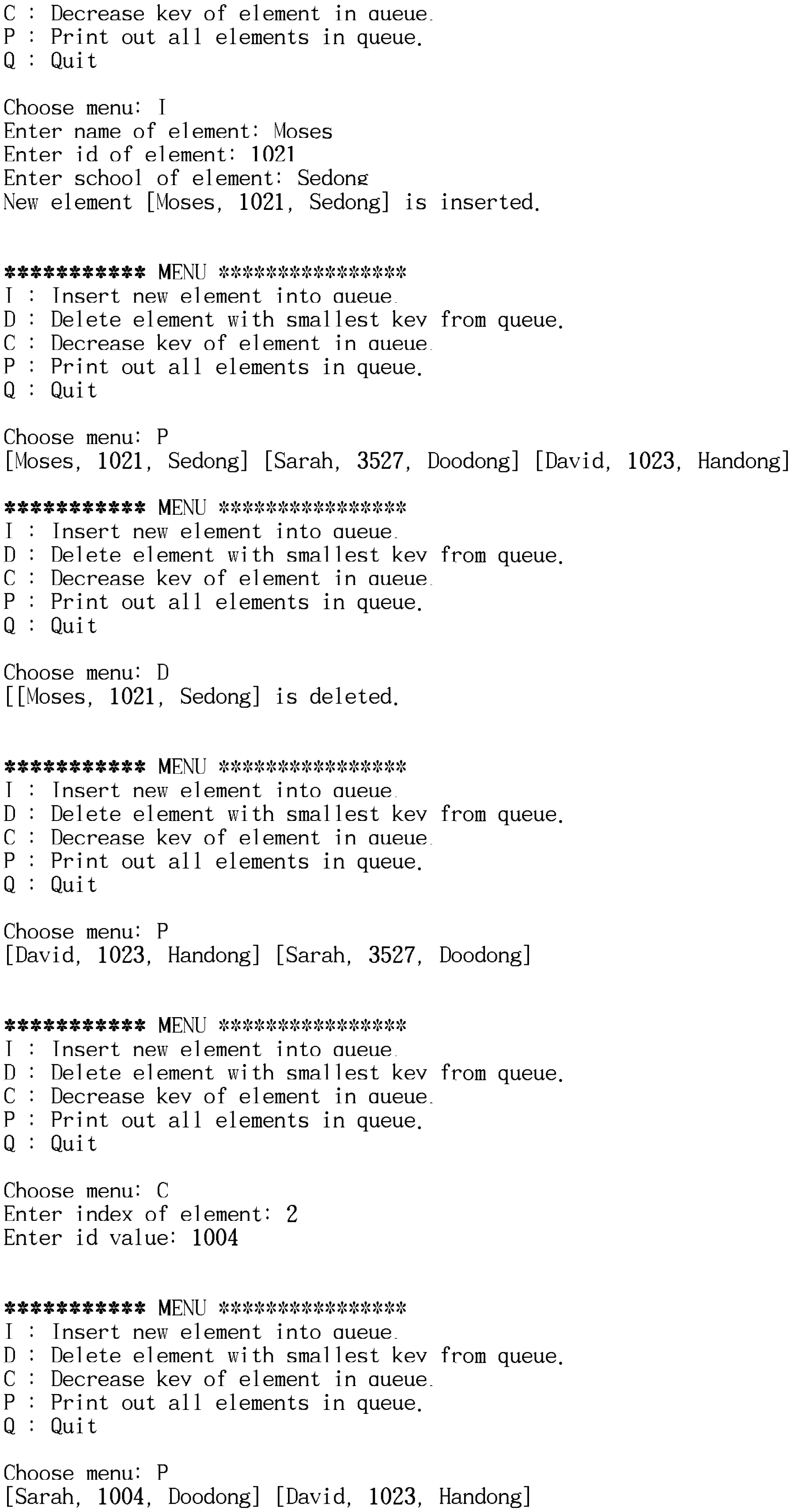
C: When this menu is chosen, program asks index of element and new score of element, then decreases key value of the element and place the element in proper position.

P: When this menu is chosen, program displays all elements in queue.

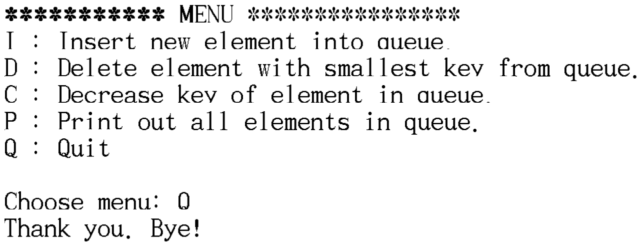
Q: When this menu is chosen, program gets terminated.

example)

```
************ MENU ****************
I : Insert new element into queue.
D : Delete element with smallest key from queue.
C : Decrease key of element in queue.
P : Print out all elements in queue.
Q : Quit

Choose menu: I
Enter name of element: David
Enter id of element: 1023
Enter school of element: Handong
New element [David, 1023, Handong] is inserted.


************ MENU ****************
I : Insert new element into queue.
D : Delete element with smallest key from queue.
C : Decrease key of element in queue.
P : Print out all elements in queue.
Q : Quit

Choose menu: I
Enter name of element: Sarah
Enter id of element: 3527
Enter school of element: Doodong
New element [Sarah, 3527, Doodong] is inserted.


************ MENU ****************
I : Insert new element into queue.
D : Delete element with smallest key from queue.
```



```
C : Decrease kev of element in aueue.
P : Print out all elements in queue.
Q : Quit

Choose menu: I
Enter name of element: Moses
Enter id of element: 1021
Enter school of element: Sedong
New element [Moses, 1021, Sedong] is inserted.


************ MENU ****************
I : Insert new element into aueue.
D : Delete element with smallest kev from queue.
C : Decrease kev of element in aueue.
P : Print out all elements in queue.
Q : Quit

Choose menu: P
[Moses, 1021, Sedong] [Sarah, 3527, Doodong] [David, 1023, Handong]

************ MENU ****************
I : Insert new element into aueue.
D : Delete element with smallest kev from queue.
C : Decrease kev of element in aueue.
P : Print out all elements in queue.
Q : Quit

Choose menu: D
[[Moses, 1021, Sedong] is deleted.


************ MENU ****************
I : Insert new element into aueue.
D : Delete element with smallest kev from queue.
C : Decrease kev of element in aueue.
P : Print out all elements in queue.
Q : Quit

Choose menu: P
[David, 1023, Handong] [Sarah, 3527, Doodong]


************ MENU ****************
I : Insert new element into aueue.
D : Delete element with smallest kev from queue.
C : Decrease kev of element in aueue.
P : Print out all elements in queue.
Q : Quit

Choose menu: C
Enter index of element: 2
Enter id value: 1004


************ MENU ****************
I : Insert new element into aueue.
D : Delete element with smallest kev from queue.
C : Decrease kev of element in aueue.
P : Print out all elements in queue.
Q : Quit

Choose menu: P
[Sarah, 1004, Doodong] [David, 1023, Handong]
```



```
*********** MENU ****************
I : Insert new element into queue.
D : Delete element with smallest key from queue.
C : Decrease key of element in queue.
P : Print out all elements in queue.
Q : Quit

Choose menu: Q
Thank you. Bye!
```

Note:

- Use class, not just structure. If you use structure only, you will get 10% deduction.
- Spaces before and/or after legitimate input should be allowed.
- Remember that homework will be graded on various inputs.

Question to think about:

- Is your program easily scalable if ‘school’ is expanded upto ‘Baekdong’?
- What if program should allow all uppercase and lowercase alphabet, such as “HanDong”, “HANDONG”, and “handonG”?
- What if ‘name’ has space between alphabet letter, such as “Sam Gould”?