January 2020 CSE208: Data Structures and Algorithms II Sessional

Offline on Advanced Data Structure - II (Binomial Heap)

In this assignment, you need to implement the Binomial (Min) Heap data structure. It must support the following operations.

1. Find-Min : Returns the node having the minimum key
2. Extract-Min : Returns the node having the minimum key and deletes it from the heap
3. Insert : Inserts a new node in the heap
4. Union : Merges two binomial heaps
5. Print : Prints the level order traversal with level no. of each of the binomial trees in the heap

Input/Output:

You will take input from a text file where each line will specify one of the aforementioned operations. The operations are denoted by the first letter i.e. 'F' indicates Find-Min operation, 'I' indicates Insert operation and so on. Then the operands will follow where necessary. You can assume that all the operands will be integers.

You have to print the output to a text file. Keep provision of printing output to the console as well (but printing both to file and to console simultaneously will not be necessary).

Check out the Sample I/O for further clarification.

Special Instructions:

Write ***readable, re-usable, well-structured, quality*** code. This includes but is not limited to writing appropriate functions for implementation of the required algorithms, meaningful naming of the variables, suitable comments where required, proper indentation etc.

Please **DO NOT COPY** solutions from anywhere (your friends, seniors, internet etc.). Any form of plagiarism (irrespective of source or destination), will result in getting -100% marks in the offline. Also, be informed that for repeated offence of plagiarism, the departmental policies suggest stricter measures.

Submission Guideline:

1. Create a directory with your 7 digit student id as its name
2. Put the source files only into the directory created in step 1
3. Zip the directory (compress in .zip format; .rar, .7z or any other format is not acceptable)

4. Upload the .zip file on moodle.

   For example, if your student id is 1705xxx, create a directory named 1705xxx. Put only your source files(.c, .cpp, .java, .h, etc.) into 1705xxx. Compress1705xxx into 1705xxx.zip and upload the 1705xxx.zip on moodle.

Failure to follow the above-mentioned submission guideline may result in upto 10% penalty.

Submission Deadline : November 13, 2020 08:00 PM

Sample I/O:

Input	Output
I 5 I 2 P I 10 P F E P U 3 20 P	Printing Binomial Heap... Binomial Tree, B1 Level 0 : 2 Level 1 : 5 Printing Binomial Heap... Binomial Tree, B0 Level 0 : 10 Binomial Tree, B1 Level 0 : 2 Level 1 : 5 Find-Min returned 2 Extract-Min returned 2 Printing Binomial Heap... Binomial Tree, B1 Level 0 : 5 Level 1 : 10 Printing Binomial Heap... Binomial Tree, B2 Level 0 : 3 Level 1 : 5 20 Level 2 : 10

Note that, the operands of Union operation indicates the keys with which you should construct a new binomial heap and then merge it with the existing one. For example, when U 3 20 is given, a binomial heap is constructed with 3 and 20 and then merged with the existing one containing 5 and 10. So the existing binomial heap is updated after an Union operation.