

CS 211 Data Structures and Algorithms Lab

Aug -- Nov, 2019

Assignment 3

Total Marks: 10

Due on 14th September

The objective of this assignment is to implement Priority Queue and Heapsort using max-heaps.

Command-line argument:

Your program should receive a file (input file) as a command line argument.

Input file

The input file will be a text file where each line will be of any of the following format:

insert <number>, maximum, extract-max, increase-key <index> <number>, sort, where <number> represents any non-negative integer.

The output must be in a file named 'heap.txt'. Every line in the input file must have a corresponding output line in priority_queue.out. The details are given below.

Command	Meaning	Output
insert <number>	Insert <number> to the priority queue	<number> inserted
maximum	Find the maximum in the priority queue	<maximum number> / <empty-line> (if the priority queue is empty)
extract-max	Find and remove the maximum from the priority queue	<maximum number> / <empty-line> (if the priority queue is empty)
increase-key <index> <number>	Make the key at <index> as <number> if <number> is at least the current value at <index>. Note that the index ranges from 0 to heap-size - 1	Key at <index> changed to <number> / <number> is less than the current key at <index>
sort	Do a heapsort on the elements in the priority queue. Note that you don't have to build a max-heap here. Further, the heap should not be disturbed. So, you may	Elements in the priority queue in ascending order. Two values are separated by a single space.

	take a copy of the heap and do the heapsort on it.	
--	--	--

Submission

- The program you submit should output heap.txt when run.
- The main file of your program should be named as <roll no>.<extension>, where roll no. specifies your roll no. and the extension depends on the language you choose (Usage of C/C++ is mandatory for this assignment). Ex: 180040001.c
- Test well before submission. We have some hidden inputs with us to test your program. The mark you obtain is purely based on whether your program correctly gives outputs for the hidden inputs.
- If your program has only a single source file, please submit the file as it is. If your program has multiple source files, please submit your code as a zip file where the name of the zip file should be your roll number. It is important that you follow the input/output conventions exactly (including the naming scheme) as we may be doing an automated evaluation. There will be a penalty of 10% (on the mark you deserve otherwise) if you do not follow the naming conventions exactly.
- Follow some coding style uniformly. Provide proper comments in your code.
- Submit only through moodle. Submit well in advance. Any hiccups in the moodle/internet at the last minute is never acceptable as an excuse for late submission. Submissions through email or any other means will be ignored.
- Acknowledge the people (other than the instructor and TA) who helped you to solve this assignment. The details of the help you received and the names of the people who helped you (including internet sources, if applicable) should come in the beginning of the main file as a comment. Copying others' programs is a serious offence and deserving penalty will be imposed if found.

Evaluation

- To consider for first evaluation without penalty, you have to submit your program by 14th September (11:59 pm). If you submit after 14th September but on or before 21st September (11:59 pm), there will be a penalty of 10% on the marks you deserve otherwise.
- If you do not submit by 21st September, your program will not be considered for the first evaluation.
- We will do the first evaluation after 21st September. The marks you obtain will be proportional to the number of correct lines in the output files. We will use the 'diff' program to check the differences between the correct output file and the output file generated by your program. So, you may verify the correctness of output file by using diff program with sample output file before submission. (See the man page of diff for more info).

- After the first evaluation, you will get a chance to improve your program. For this, after modification, you can submit your code for second evaluation. It comes with a 20% penalty. The due date for the second evaluation will be announced after the first evaluation. Those who submit their code after 21st September and before the due date for second evaluation will also be considered for the second evaluation. Submission done after the due date of the second evaluation will be ignored.