

## CSE 331 Homework 4 (Written problems due at start of class on Wednesday March 1)

Programming part is due by 11:59 pm EST Sunday March 5

1. Prove that an  $n$ -element heap has height  $h = \lfloor \log_2 n \rfloor$ . You need to prove this for any given  $n$ . Showing  $h = 3$  for  $n = 8$  is not a proof. (5 points)
2. Apply `deleteMin` to the heap  $H[] = \{2, 3, 7, 4, 9, 8, 15, 13, 10\}$  using the pseudocode in Weiss 2014 chapter 6.3 (see Figure 6.12). Show the contents of the **array**  $H$  whenever there is a change. (3 points)
3. Insert key 1 to the following heap  $H$  using the pseudocode in Weiss 2014 chapter 6.3 (see Figure 6.8). Show the contents of the **array**  $H$  whenever there is a change.  $H[] = \{2, 3, 7, 4, 9, 8, 15, 13, 10\}$ . (4 points)
4. This problem consists of two parts. *First*, give an  $O(n \log_2 k)$ -time algorithm to merge  $k$  sorted lists into one sorted list, where  $n$  is the total number of elements in all the input lists. (*Hint*: Use a min-heap for  $k$ -way merging.) Note, if you need to use any defined operation on a heap, just treat it as a function and don't need to worry about the details. For example, if you want to build a heap on an array  $A$ , just say `buildHeap(A)`. However, you need to describe on what elements you build this heap. For your designed algorithm, *analyze its time complexity*.

In the *second* part of this problem, show the contents and changes of the heap you built in order to merge three sorted lists  $[0, 1, 3]$ ,  $[4]$ , and  $[2, 5]$ . (15 pts)

5. This problem is a programming problem. Implement a program which builds a heap for any input array using the algorithm described in

Weiss 2014 chapter 6.3 (see Figure 6.14). A test program is provided in the attached file `heap-shell.cpp`.

You can use other programming languages. But you must implement the same framework. Your program needs to ask users to input a number. And the array will need to be filled by random numbers. And you need to follow the same type of output format as the given test program. Only submit your **source codes and a readme.txt file** that explains how to compile and run your program. All source code files should be named with a prefix `heap-shell`. For example, if you implement your program in a single C++ source code file, the file should be named `heap-shell.cpp`. We will compile your program on `arctic.cse.msu.edu`. Please make sure it can compile and run correctly on `arctic`. For any readme file, please use `.txt` format. **Don't use doc, pdf, ps, zip, docx.** 10 pts for compile. 20 pts for correct implementation. (30 pts)