

# CS240, Spring 2022

## Assignment 2: Question 4

**Q4a)** On the example  $(15, 10, \dots)$  above, show us what these heaps would contain at each of the 5 steps (we don't know if these are min-heaps or max-heaps yet, so just tell us what elements they contain).

Steps	$H_{lo}$	$H_{hi}$
1	15	
2	10	15
3	1, 10	15
4	1, 10	15, 20
5	1, 10, 15	20, 30

**Q4b)** We would like to be able to read off the (current) median using just one access to  $H_{lo}$ . What kind of heap should it be, a min-heap or a max-heap? How long does finding the current median take?

We should use a max-heap, finding the current median should take  $O(1)$  time as we are accessing the root of  $H_{lo}$

**Q4c)** Describe how to update the two heaps when inserting the next element. In particular, in which heap do you insert the element, and how do you ensure that  $H_{lo}$  and  $H_{hi}$  have the required size afterwards? Give the runtime of your update method, with a short justification; it should be  $o(n)$ . (At this stage, you will have to explain whether  $H_{hi}$  should be a min-heap or a max-heap.)

We know that the maximum of  $H_{lo}$  and the minimum of  $H_{hi}$  is stored in the roots. We can check if the new element is greater then the max of  $H_{lo}$  we can insert it into  $H_{hi}$ . If element is less then the minimum of  $H_{hi}$  we can insert into  $H_{lo}$

If the counter for  $H_{lo}$  is larger then  $H_{hi}+1$  then we can use deleteMax for  $H_{lo}$  after storing the value in a temporary value, and vise versa for if  $H_{hi}$  is larger then  $H_{lo}+1$ . We will then apply a fix down to make sure the order remains the same

Note that our inserting, deleting (min/max) will take  $O(\log(n))$  and from our lectures we know that  $\log(n) \in o(n)$ . Thus it follows that our algorithm will have a time complexity of  $o(n)$  as required.