Trees: Heaps/Priority Queues

April 8, 2019

1 Max -Heap

Consider the array A = {29, 18, 10, 15, 20, 9, 5, 13, 2, 4, 15}

i. Does A satisfy the max-heap property? If not, fix it by swapping elements (5 pts)

ii. Using array A (possibly corrected), illustrate the execution of the heap-extract-max algorithm, which extracts the max element and then rearranges the array to satisfy the max-heap property. For each iteration or recursion of the algorithm, write the content of the array A (5 pts).

2 Min-Heap

2.1

Consider the array: A = {4, 33, 6, 90, 33, 32, 31, 91, 90, 89, 50, 33 }

i. Is A a min-heap? Justify your answer by briefly explaining the min-heap property (5 pts).

ii. If A is a min-heap, then extract the minimum value and then rearrange the array with the min-heapify procedure. In doing that, show the array at every iteration of min-heapify. If A is not a min-heap, then rearrange it to satisfy the min-heap property (5 pts)

2.2

Draw the binary min heap that results from inserting: 77, 22, 9, 68, 16, 34, 13, 8 in that order into an initially empty binary min heap. You do not need to show the array representation of the heap. You are only required to show the final heap, although if you draw intermediate heaps, please circle your final result for ANY credit (10 pts).

Once your heap is complete, answer the following question.

i. Draw the binary min heap that results from doing 2 deletemins on the heap you created in part a). You are only required to show the final heap, although if you draw intermediate heaps please circle your final result for ANY credit (5pts).