CSCI 470 Fundamentals of Algorithms
Howard University
Instructor: Vijay Chaudhary — Homework 2

Homework 2

Please write your solutions in the LaTeX. You may use online compiler such as Overleaf or any other compiler you are comfortable with to write your solutions in the LaTeX.

Due date: Monday Oct 02, 9:10 AM EDT [FINAL DEADLINE].

I will collect your submission when the class meets on Monday, Sep 25. Please handover a printed copy of your Homework 2 solutions (preferably written in the LaTeX). Also, please make sure that you have your full name and student ID in your submission.

You can use the LaTeX submission template I have shared along with the homework. There are two .tex files ("macros.tex", and "main.tex"). You can upload the zipped folder directly to Overleaf or create a blank project on Overleaf and upload macros.tex and main.tex files, and edit main.tex to write your solutions. "macros.tex" is mostly for macros (predefined commands).

Handwritten solutions will also be accepted. Points will be deducted if handwritten solutions are not legible.

Problem 2-1. Starting with procedure MAX-HEAPIFY,

(a) (10 points) write pseudocode for the procedure MIN-HEAPIFY(A, i), which performs the corresponding manipulation on a min-heap.

(b) (5 points) State the running time of MIN-HEAPIFY. How does this running time of MIN-HEAPIFY compare to that of MAX-HEAPIFY?

Problem 2-2. (5 points) Why do we want the loop index i in line 2 of BUILD-MAX-HEAP to decrease from $\lfloor A.length/2 \rfloor$ to 1 rather than increase from 1 to $\lfloor A.length/2 \rfloor$?

BUILD-MAX-HEAP(A)

```
1  A.heapsize = A.length
2  for i = ⌊A.length/2⌋ downto 1
3      MAX-HEAPIFY(A, i)
```

Pseudocode of BUILD-MAX-HEAP(A) *above for reference.*

Problem 2-3.

HEAPSORT(A)

```
1  BUILD-MAX-HEAP(A)
2  for i = A.length downto 2
3      exchange A[1] with A[i]
4      A.heapsize = A.heapsize − 1
5      MAX-HEAPIFY(A, 1)
```

(10 points) Argue the correctness of HEAPSORT using the following loop invariant:

> At the start of each iteration of the **for** loop of lines 2-5, the subarray $A[1 \,..\, i]$ is a max-heap containing the i smallest elements of $A[1 \,..\, n]$, and the subarray $A[i+1 \,..\, n]$ contains the $n - i$ largest elements of $A[1 \,..\, n]$, sorted.

Make sure that your loop invariant fulfills the three necessary properties.

Problem 2-4. (20 points) Write pseudocode for the following procedures in order to implement a min-priority queue with a min-heap. You may assume that MIN-HEAPIFY is already provided to you to use.

(a) (2 points) HEAP-MINIMUM

(b) (10 points) HEAP-EXTRACT-MIN

(c) (10 points) HEAP-DECREASE-KEY

(d) (8 points) MIN-HEAP-INSERT

Problem 2-5. (10 points) Show that the running time of QUICKSORT is $\Theta(n^2)$ when the array A contains distinct elements and is sorted in decreasing order.

Extra Credit

Problem 2-6. (10 points) What is the running time of HEAPSORT on an array A of length n that is already sorted in increasing order? What about decreasing order?

Problem 2-7. (10 points) Show that there are at most $\lceil n/2^{h+1} \rceil$ nodes of height h in any n-element heap.

Problem 2-8. (10 points) Suppose that the splits at every level of quicksort are in the porportion $1-\alpha$ to α, where $0 \leq \alpha \leq 1/2$ is a constant. Show that the minimum depth of a leaf in the recursion tree is approximately $-\lg n / \lg \alpha$ and the maximum depth is approximately $-\lg n / \lg(1-\alpha)$.