CS 240: Data Structures and Data Management **Fall 2022**

Midterm Practice Problems

Note: This is a sample of problems designed to help prepare for the midterm exam. These problems do *not* encompass the entire coverage of the exam, and should not be used as a reference for its content.

1 True/False

Indicate "True" or "False" for each of the statements below.

a) If $T_1(n) \in \Omega(f(n))$ and $T_2 \in O(g(n))$, then $\frac{T_1(n)}{T_2(n)} \in \Omega\left(\frac{f(n)}{g(n)}\right)$.

b) If $f(n) \in O(g(n))$ and $g(n) \in \Omega(h(n))$, then $f(n) \in \Omega(h(n))$.

c) If $\lim_{n\to\infty} \frac{f(n)}{g(n)} = e^{42}$, then $f(n) \in \Theta(g(n))$.

d) If $f(n) \in o(n \log n)$, then $f(n) \in O(n)$.

e) The average-case and expected-case run-time of an algorithm must always be the same.

f) In a max-oriented binary heap, for some node i, all nodes with index less than or equal to $\lfloor \frac{i-1}{2} \rfloor$ have a value greater than or equal to that of i.

g) All heaps satisfy the AVL height-balance requirement.

h) A binary search tree with n leaves must have height in $O(n)$.

2 Order Notation

a) Show that $3n^2 - 8n + 2 \in \Theta(n^2)$ from first principles.

b) Show that $2^n \in \omega(n^2)$.

c) Complete the statement $n! \in \sqcup(n^n)$ by filling in $\sqcup$ with either Θ, o, or ω, and prove the corresponding relationship.

d) Complete the statement $\frac{n^2}{n+\log n} \in \sqcup(n)$ by filling in $\sqcup$ with either Θ, o, or ω, and prove the corresponding relationship.

e) Prove or disprove: if $f(n) \in \Theta(g(n))$, then $\log f(n) \in \Theta(\log g(n))$. Assume that $f(n)$ and $g(n)$ are positive functions. You should prove the statement from first principles or provide a counter example.

3 Runtime Cases

After midterm grading is complete, Angel wants to make sure all the grades are okay, and have them listed in sorted order. She asks Yundi or Prashanth to help with this task. Yundi and Prashanth have many sorting algorithms to choose from, so their decision is based on their mood. Their moods fluctuate while they check assignment grades, and eventually depends on the final student's grade.

Assignment grades in array A are integers ranging from 0 to 100. There is exactly one student with a grade of 0, and exactly one student with a grade of 100. The function `verify(A)` runs in $\Theta(n)$ time and returns:

- +1, if the final student of A scored 100.
- -1, if the final student of A scored 0.
- 0, otherwise.

```
YundiSort (A):
    mood := verify(A)
    if mood = +1
        MergeSort(A)
    else if mood = -1
        SelectionSort(A)
    else MSDRadixSort(A, R = 10, m = 3)
```

```
PrashanthSort (A):
    mood := verify(A)
    if mood = +1
        SelectionSort(A)
    else if mood = -1
        LSDRadixSort(A, R = 10, m = 3)
    else MergeSort(A)
```

a) What is the worst-case, and average-case runtime for `YundiSort`?

b) What is the worst-case, and average-case runtime for `PrashanthSort`?

c) Angel's decision, which we refer to as `AngelSort`, is to flip a coin. If it flips heads, she gets help from Yundi (`YundiSort`). Otherwise, if it flips tails, she gets help from Prashanth (`PrashanthSort`). What is the expected runtime (worst-case expected) for `AngelSort`?

4 Pseudocode Runtime Analysis

Analyze the worst-case runtimes for the following pseudocodes as functions of n. A $\Theta()$ bound is sufficient.

a)

```
j ← 0;
k ← 1;
while j ≤ n do
    j ← j + k;
    k ← k + 2;
end
```

b) For the following algorithm, you may assume that n is a power of 3.

Algorithm 1: STOOGE(A, i, j)

Input: Array A of size n, index i (initially 0), index j (initially $n-1$)

Output: No output but the subarray $A[i \dots j]$ will be sorted

```
if A[j] < A[i] then
    SWAP(A[i], A[j])
end
if j − i + 1 > 2 then
    t ← ⌊(j−i+1)/3⌋;
    STOOGE(A, i, j − t);
    STOOGE(A, i + t, j);
    STOOGE(A, i, j − t);
end
```

c)

```
x ← n;
while x > 1 and x < n^12 do
    if x is even then
        x ← x/2;
    end
    x ← 3x + 1;
end
```

5 Numbers in Range

We have an array $\mathcal{A}$ of n non-negative integers such that each integer is less than k, where k is a positive integer. Give an algorithm with $O(n+k)$ preprocessing time such that queries of the form "how many integers are there in $\mathcal{A}$ that are in the range $[a, b]$?" can be answered in $O(1)$ time.
Note that a and b are not fixed; they are parameters given to the query algorithm.

6 Multi-way Merge

Given a set of k sorted arrays, where the combination of the k arrays has n elements in total, design an $O(n \log k)$ algorithm that produces a single sorted array containing all n elements.
Hint: use a priority queue.

7 d-ary Heaps

Suppose instead of binary heaps, we have d-ary heaps, where each internal node contains d children, except possibly the last one.

a) What is the height of a d-ary heap of n nodes?

b) Suppose the d-ary heap is represented in an array similar to binary heaps. For a node at index i, give the indices of its parent and all of its children.

c) Give an efficient algorithm for `Insert` and analyze its runtime.

d) Given an efficient algorithm for `deleteMax` and analyze its runtime.