CSci 242: Algorithms and Data Structures

Instructor: Dr. M. E. Kim Date: November 15th, 2018

Exam 2: 100 points + 10 (Optional)

Name: _____

1. Your answer should be precise and fully described; any sloppy answer will not get a full point.
2. Your hand writing should be **clear, dark, large enough (≥ 9 pt.)** and **readable**.

Mark the followings;

Difficulty:

Very Easy: _____ Easy: _____ Moderate: _____ Difficult: _____ Very Difficulty: _____

Time:

Short: _____ Enough: _____ Too Much: _____

Comment:

Q1. [30] **Short Answer**

Explain your answer clearly.

1) [5] Give the name of one of the sorting algorithms whose running time is **O(*n*)**.

2) [5] Give the names of sorting algorithm which was designed based on ***Divide & Conquer paradigm.***

3) [10] Give the ***recurrence equation*** for the running time of Quick Sort algorithm both (A) in the ***worst*** case and (B) in the ***best*** case. Then, (C) give their solutions of the running time in Big-Oh (**O**) notation.

4) [5] Describe the algorithm design paradigm of ***Divide and Conquer***.

5) [5] In the data encoding/decoding of the characters, what is/are the least requirement to achieve the ***optimal codes with no ambiguity***?

Q2. [20] **Job Scheduling Problem**

Suppose a hair stylist has several customers waiting for different treatments. The treatments don't all take the same amount of time, but the stylist knows how long each takes. A reasonable goal would be to schedule the customers in such a way as to minimize the total time they spend both waiting and being served, which is called the time in the system.
This is called *a problem of minimizing the total time in the system.:* total time = waiting time + service time.

Five customers and their service times are given below:

Customer	Service Time (min.)
1	40
2	20
3	80
4	50
5	60

1) [10] Write a recursive **or** an iterative greedy algorithm, **Schedule(Customer, ?)**, that both decide ***the optimal sequence*** of customers to minimize the total time spent in the system and also computes ***the total system time*** in the system.
An input is given in the array named Customer[1 .. N] in which Customer[*i*] stores a service time of customer-*i*.
You can define more arguments '?' of the algorithm if they're needed.

2) [10] (A) What is the ***optimal solution*** of the above problem, i.e. the optimal sequence of customers? and (B) What is its ***minimum total system time***?

Q3. [20] **Divide & Conquer**

1) [10] Write a in-place recursive algorithm, named **Minimum(A, *a, b*)**, based on the Divide & Conquer paradigm to find the minimum element in the array A[*a* .. *b*]. Suppose the number of elements stored in A is $n = 2^k$, i.e. it's always dividable by 2.

2) [5] Write a ***recurrence equation*** for the running time of the algorithm in 1).

3) [5] By Master's Theorem, give the ***solution*** of the recurrence in Big-Theta (**Θ**)

Q4. [20] **Huffman Codes**

1) [10] The text file contains only the characters shown in the table with the given frequency. Generate the ***optimal Huffman code*** for each character in the table.

character	A	I	O	Y	D	H	L	P	!	space
frequency	24	30	12	4	18	20	10	16	1	25
code										

2) [5] What is the ***total number of bits*** required to encode the text file using your Huffman codes?

3) [5] Decode the following codes into a text.

100101010010111101110100111011111000111011111101111100

Q5. [10] **Recurrence**

For a given recurrence equation below,

$$T(n) = \begin{cases} 1 & n < 3 \\ 2T(n/3) + n & n \geq 3 \end{cases}$$

1) [10] Solve it by ***Master's Theorem***. Clearly state the case to which it belongs, the rationale of the solution and its solution in the big-Theta(Θ) notation.

2) [10, optional] Solve it **either** (2A) by Recursion Tree method **or** (2B) by Iterative Substitution method.

(2A) In Recursion Tree, draw it by specifying:

(a) the height of tree, (b) the number of leaves, level, (c) the number of nodes per level, (d) the size of input per level, (e) a per-level time, (f) the total time **and** (g) its asymptotic tight bound (i.e. the asymptotic tight bound of the solution in the big-Theta),

(2B) In Iterative Substitution:

The proper number of iterative steps and the final step where the size of input is 1, the computation of some parameters.