CRAWFORD UNIVERSITY, FAITH CITY, IGBESA

COLLEGE OF NATURAL AND APPLIED SCIENCES

DEPARTMENT OF COMPUTER AND MATHEMATICAL SCIENCES

HARMATTAN SEMESTER **SESSION: 2022/2023**

COURSE CODE: CSC 416 **COURSE TITLE: ALGORITHM AND COMPLEXITY**

UNITS: 3 **TIME: 2:30HOURS**

ANSWER ANY FOUR (4) QUESTIONS

1a. Write a liner searching algorithm to search for a given customer number called 0809789999 in a set of numbers of n size. The algorithm should return 1 if the number is found or otherwise the algorithm should return 0. 6marks

b. List and give the formal definition of the three (3) asymptotic notion mostly used to measure time complexity 9marks

2a.

We can extend our notation to the case of two parameters n and m that can go to infinity independently at different rates. For a given function $g(n, m)$, we denote by $O(g(n, m))$ the set of functions

$$O(g(n,m)) = \{ f(n,m) : \text{there exist positive constants } c, n_0, \text{ and } m_0 \text{ such that } 0 \le f(n,m) \le cg(n,m) \text{ for all } n \ge n_0 \text{ or } m \ge m_0 \} .$$

Give corresponding definitions for $\Omega(g(n, m))$ and $\Theta(g(n, m))$.

10marks

b. Show that the solution of $T(n) = T(n-1) + n$ is $O(n^2)$ 5marks

3a. Give the asymptotic upper and lower bounds of the following recurrences

$T(n)=\sqrt{n}\, T(\sqrt{n}) + n$ 5marks

b. Analyze the cost and time of the algorithm below

```
INSERTION-SORT(A)
for j = 2 to A.length
    key = A[j]
    // Insert A[j] into the sorted
        sequence A[1..j − 1].
    i = j − 1
    while i > 0 and A[i] > key
        A[i + 1] = A[i]
        i = i − 1
    A[i + 1] = key
```

10marks

4a. Write a quick sort recursive algorithm 5marks

b. Using substitution method, find the running time of the following recurrence equations 4marks

$T(n) = 2T(n/2) + n^2$

c. Write an algorithm to find (n-1)! using recursive method 6marks

5a. State and write out all the rules of master theorem 6marks

b. List the hierarchy of the type of algorithm 3marks

b. Using master theorem, find the running time of the function below

i. $T(n)= 2T(n/2) + 3$ ii. $T(n) =4T(n/2) + \log^2 n$ iii. $T(n) = 16T(n/4) + n^2 \log n$ 6marks

6a. Explain the term "divide-and-conquer" 3marks

b. Write a merge sorting recursive algorithm 5marks

c. For a concrete example, let us pit a faster computer (computer A) running insertion sort against a slower computer (computer B) running merge sort. They each must sort an array of 10 million numbers. (Although 10 million numbers might seem like a lot, if the numbers are eight-byte integers, then the input occupies about 80 megabytes, which fits in the memory of even an inexpensive laptop computer many times over.) Suppose that computer A executes 10 billion instructions per second (faster than any single sequential computer at the time of this writing) and computer B executes only 10 million instructions per second, so that computer A is 1000 times faster than computer B in raw computing power. To make the difference even more dramatic, suppose that the world's craftiest programmer codes insertion sort in machine language for computer A, and the resulting code requires 2n2 instructions to sort n numbers. Suppose further that just an average programmer implements merge sort, using a high-level language with an inefficient compiler, with the resulting code taking 50n lg n instructions. To sort 10 million numbers, computer A takes 7marks

Best of luck!!!!!!