National University of Computer and Emerging Sciences
Karachi Campus

Design and Analysis of Algorithms (CS2009)

Date: Sep 23rd 2024

Course Instructor(s)

Dr.Muhammad Atif Tahir, Dr.Nasir Uddin, Dr.Kamran Ali, Dr. Fahad Sherwani, Dr. Farrukh Salim Ms.Anaum, Mr.Syed Faisal Ali, Mr.Sandesh, Mr.Minhal Raza, Mr. Abu Zohran

Mid-1 Exam

Total Time (Hrs):	1
Total Marks:	12.5
Total Questions:	3

Do not write below this line

Attempt all the questions.

CLO #1: To apply acquired knowledge to solve computing problems complexities and proofs

---

Question 1:

(a): Solve the following recurrence relations by using the **Master Theorem** **[1.5 marks]**

a) $T(n) = 100T\left(\frac{n}{10}\right) + n^2 \log n + n^2 + 1$

b) $T(n) = 2T\left(\frac{n}{4}\right) + 3T\left(\frac{n}{2}\right) + n$

c) $T(n) = 3T\left(\frac{n}{3}\right) + \sqrt{n} + 1$

(b): Solve the following recurrence relations by using the **Guess & Test Method** **[1 marks]**

a) $T(n) = 3T\left(\frac{n}{2}\right) + n^2$; apply for (i) $T(n) = O(n^2)$ and (ii) $T(n) = O(nlogn)$

(c): Solve the following recurrence relations by using the **Recursion Tree or Iterative Method** **[3 marks]**

a) $T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + n$

b) $T(n) = 2T\left(\frac{n}{2}\right) + n^2$

(d): Prove using following **loop invariant** that the following function correctly sums the elements in the array passed to it.

[1.5 marks]

Loop Invariant: At the start of each iteration of the loop, sum is the sum of the first i elements of A, and i is an integer such that $0 <= i <= n$.

```
def sum(A, n):
    sum = 0
    i = 0
    while i < n:
        sum = sum + A[i]
        i = i + 1
    return sum
```

CLO #2: To analyze complexities of different algorithms using asymptotic notations, complexity classes and standard complexity function

Question 2: Prove the following statement True or False

[1.5 marks]

1. $n^3 + 2^n = O(2^n)$
2. $\frac{n^2+4}{2n^2+3n+1} = \theta(1)$
3. $n^2 logn = \theta(n^2)$

CLO #4: To construct and analyze real world problems solutions using different algorithms design techniques

Question 3:

Suppose you are given an array A with n entries, with each entry holding a distinct number. You are told that the sequence of values $A[1], A[2], \ldots, A[n]$ is unimodal: For some index p between 1 and n, the values in the array entries increase up to position p in A and then decrease the remainder of the way until position n. You'd like to find the "peak entry" p without having to read the entire array in fact, by reading as few entries of A as possible. Show how to find the entry p by reading at most $O(log\ n)$ entries of A.

[4 marks]