SUBJECT: **Data Structures & Algorithm Design**   CODE**: CS 6103**

FULL MARKS: 40 +10   Duration of Examination**: 2 Hours**

8th December 2021(Friday) 3-5PM

[Total three questions to be answered]

[Executive PhD Students have to answer at least one questions each from section A& B]

[Other Students have to answer any three questions from section A]

All question should be answered by own hand writing and uploaded to MS team

[Start time 2.50PM, Finish time 4.30PM, Upload on or before 4.45PM]

[Quiz Start Time-4.50PM, Close Time 5.20PM]

Instruction:

- The examination is hand written close book examination, followed by a Quiz test [MCQ], with MS team (code **y1npi6v**)

- Write your Name and Roll number clearly in top of your answer sheet.

- You have to upload the scan copy of your hand written answer sheet (pdf format only) in MS team (appropriate assignment) on or before **4.45PM.**

- Rename your submission file as **Rollno_DSAD**; the student with roll no 117CS0246 has to rename the answer file for the Question as **117CS0246_DSAD**.

- Call me on 9937324437 or 9337938766 for any assistance during the examination. My E-mail id is bdsahu@nitrkl.ac.in, bibhudatta.sahoo@gmail.com.

SECTION A

1[a]   Assume a hash table with capacity M=31 gets to be over 3/4ths full. We decide to *rehash. What is a good size choice for the new table to reduce the load factor below* 0.5 and also avoid collisions?

1[b]   Compute a table representing the KMP failure function for the pattern "amalgamation".

1[c]   Define and differentiate between polynomial-time approximation scheme and fully polynomial-time approximation scheme with appropriate example?

1[d]   Explain the basic advantages of dynamic Hashing over static hashing?

1[e]   What are the aadvantages of metaheuristics techniques over the classical optimization methods

1[f]   Name the main features of Genetic Algorithms (GA). What two requirements should a problem satisfy in order to be suitable for solving it by a GA?

[6×2]

2[a]   Explain the basic concept of "dynamic hashing without directories"? Demonstrate the action of directory less dynamic hashing for the given set of key values {16, 4, 6, 8, 21, 10, 32, 7, 9, 20, & 26} in a hash table with bucket size = 4 and Hash Function that returns LSBs of a given key "X".

2[b]   What are the different approaches to prove the correctness of an algorithm Explain Loop Invariants associated with an algorithm? Use binary search algorithm to discuss the process of proving correctness?

[6+6]

3[a]   Write an algorithm to sort a "sorted-array of integers" in the reverse order? Explain the steps to prove correctness of your algorithm.

3[b]   Define the traveling salesman problem (TSP) as optimization problem? Prove that traveling salesman problem (TSP) is NP Hard?

[6+6]

4[a]   Given the values {241, 424, 289, 430, 222, 397, 392}, a hash table of size 7, and hash function $h(x) = x \bmod 7$, show the resulting tables after inserting the values in the given order with each of these collision strategies: separate chaining, linear probing, quadratic probing, and double hashing with hash function $h'(x) = (2x - 1) \bmod 7$.

4[b]   What are the different approaches to prove the correctness of an algorithm? Explain Loop Invariants associated with an algorithm? Discuss the process of proving correctness with reference to an algorithm.

[6+6]

5[a]   Define lower bound of a problem? What is the difference between worst case lower bound and average case lower bound? Find out the lower bound of heap sort algorithm.

5[b]   Discuss the concept of pattern matching algorithm? Present the Boyer-Moore algorithm? What is the advantage of Boyer-Moore method over Brute Force method to match a pattern P in a given string T. Justify the role of function last(c) to improve the performance of Boyer-Moore algorithm?

[6+6]

6[a]   Explain with the aid of a simple example, one open addressing and one alternative method that will overcome the problem of collisions within hash tables. Compare the two methods you have described, taking into account time complexity analysis for inserting and searching, and explain for each, how the hash table would be filled.Explain what is meant by rehashing and when it is appropriate to use such a mechanism.

6[b]   Why should an approximation algorithm be polynomial? What are the main steps for designing an approximation algorithm? How does lower bound of a problem play role in deriving approximation ratio? Present the definition of an approximation ratio to measure the approximation quality of the greedy algorithm.

[6+6]

=================== END OF SECTION A ===================

SECTION B

(This section is for Executive PhD Students. Answer at least one question from section B]

8[a] Explain what are randomized algorithms? What is the main difference between *Las Vegas* and *Monte Carlo* algorithms?

8[b] Write an algorithm that sorts an array $A[1..n]$ of integers in the range 1..1000 in descending (i.e., larger numbers come first) order. Give the asymptotic running time of your algorithm in terms of number of array accesses or some other realistic metric.

8[c] Define a problem class P & NP? What is reducibility in the context of NP-completeness?

8[d] Write a Sherwood-type sorting algorithm?

8[e] How does the Dynamic Programming paradigm differs from the Greedy paradigm?

8[f] Define *subset paradigm* and *ordering paradigm* in the context of greedy approach? What is the worst case space requirement in the 0/1 knapsack problem?

[6×2]

9[a] Why do we analyze a algorithms? How can we analyze algorithms? Explain the differences between average-case running time analysis and expected running time analysis of an algorithm. For each type of running time analysis, name an algorithm we studied to which that analysis was applied.

9[b] Write a randomized algorithm to find a minimum-spanning tree for undirected graph. What is the time complexity of these algorithms? Explain how representation of the graph affects complexity measure?

[6+6]

10[a] Give the algorithm of Binary search. Explain how it functions? Devise a ternary search algorithm that first tests the element at position n/3 for equality with some value x, and then checks the element at 2n/3 and either discovers x or reduces the set size to one-third the size of the original. Compare this with binary search?

10[a] What are the four complexity classes involving randomized algorithms? Suggest an randomized algorithm to compute the value of π . What is the complexity of the algorithm?

[6+6]

11[a] Show that the expected running time of randomized quick sort algorithm RandQS on n numbers is O(n log n).

11[b] Define a non-deterministic algorithm? Write a non-deterministic algorithm to find the index of a maximum element in a list of *n* elements. Discuss its complexity class with reference to randomized algorithm?

[6+6]

----------------------Submit your answer on or before 4.45PM-----------------