Ramakrishna Mission Vivekananda Educational and Research Institute

Belur Math, Howrah, West Bengal
Department of Computer Science
Midterm Examination
MSc DSAI

Course: CS110 (Data Structures and Algorithms) — Date: 22 Sep 2025
Instructor: Sri Niladri Banerjee — Time: 2 hrs
Student Roll: — Max Marks: 60
Student Name:

Instructions:

- Answers should be clear, concise, and complete to earn the full allotted marks. All steps in the problem-solving process should be clearly stated.
- Pseudocode should be close to C/C++/Python.

Section A **[10 questions × 2 marks = 20 marks]**

---

1. What are the key differences between a compiled language and an interpreted language? Provide an example for each and discuss their advantages and disadvantages.
2. Define a bit and a byte. If a computer uses 8-bit bytes, how many unique unsigned integer values can a single byte store? Using a diagram, illustrate how the value 7 would be stored in a byte of memory. Assume `int` consumes 2 bytes.
3. Write down the characteristics of an algorithm.
4. Arrange the functions in ascending order of their time complexity: $2^n,\ n!,\ n^{\log n}$
5. What is Big Θ notation? Provide a diagram and an example.
6. Explain the Addition property of Big O notation and provide a proof.
7. Find the time complexity of the following loop:
```
for (i = n; i >= 5; i = $\sqrt{i}$)
    x = x + 1;
```
8. Find the time complexity of the following loop:
```
while n > 1 do
    for i = 1 to n do
        x = x + 1;
    end for
    n = floor(n / 2);
end while
```
9. We have an array `int A[-10..10][-20..20]`. Width of `int` = 2 bytes. Base Address = 1000. Find the address of `A[5][6]` in both Row Major Order (RMO) and Column Major Order (CMO).
10. Write the `pop` functionality of a singly linked list. Assume you pop from the tail. You have both head and tail pointers.

Section B [5 questions × 4 marks = 20 marks]

1. Solve the recurrence relation:

$$T(n) = \begin{cases} 1, & \text{if } n = 0, \\ T\left(\frac{n}{2}\right) + T\left(\frac{2n}{5}\right) + 7n, & \text{if } n > 0. \end{cases}$$

2. Consider an array $A = \{3, 8, 2, 5, 7, 6, 12\}$ of length 7.
   - A subarray is a sequence of consecutive elements in the original array.
   - A subarray of size w contains exactly w consecutive elements.

   Let $w = 4$. Valid subarrays: {3,8,2,5}, {8,2,5,7}, {2,5,7,6}, {5,7,6,12}. Their sums are

$$3 + 8 + 2 + 5 = 18, \quad 8 + 2 + 5 + 7 = 22, \quad 2 + 5 + 7 + 6 = 20, \quad 5 + 7 + 6 + 12 = 30.$$

   The maximum sum is 30. Write an optimal algorithm (time complexity $< n^2$) to find the maximum sum of any subarray of size w.

3. Design an algorithm to find the middle node of a singly linked list without using its length. Examples:

   10 (*head*) → 20 → 30 → 40 → 50 (*tail*) → `None` (middle node = 30)
   10 (*head*) → 20 → 30 → 40 → 50 → 60 (*tail*) → `None` (middle node = 40)

4. Write an algorithm to reverse a singly linked list.

   Original : 10 (*head*) → 20 → 30 → 40 → 50 (*tail*) → `None`

   Reversed : `None` ← 10 (*tail*) ← 20 ← 30 ← 40 ← 50 (*head*)

5. Write a recursive algorithm to compute sin(x) using the series. Show the recursion tree starting with count = 3.

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

   Assume x in radians; terminate recursion by term count.

Section C [2 questions × 10 marks = 20 marks]

1. Illustrate with diagram the Tower of Hanoi problem for $n = 1, 2, 3$ disks, showing the sequence of moves. Write a recursive algorithm to illustrate the Tower of Hanoi problem. Determine its time complexity and why it becomes impractical for large n.

2. Let `int A[15]={0,1,2,...,14}` and key = 5.
   (i) Write both iterative and recursive binary search algorithms; show each step in a table.
   (ii) Compute average times for successful and unsuccessful search.