ourdiu.com | Shared by Aszak Ahmed Fahim

Daffodil International University

Faculty of Science & Information Technology

Department of Software Engineering

Midterm Examination, Fall 2026

Course Code: SE 131, Course Title: Data Structure

Level: 2 Term: 1 Section: A, B, C, D, E, F, G

Instructor: AB (A, B), MAK (C), TRN (D, E, F), ABS (G)

One and a half hours (1:30 Hrs) **Marks: 25**

Directions:

- **Answer questions serially.**

1. A.	**Describe** linear and non-linear Data Structure with examples.	[CLO-1, Level-1] [Marks-2+2]
B.	**Define** the following terms: I. Stack II. Linked List III. Recursion IV. Iteration	
2. A.	A teacher stores student ID numbers in an unsorted array. **Explain** the appropriate searching algorithm to find a particular student ID and also **explain** why it is suitable.	[CLO-2, Level-2] [Marks-1+3]
B.	Consider the sorted array of student IDs: [12, 25, 38, 46, 59, 67, 81, 94]. **Calculate** the number of comparisons (complexity) required by Linear Search and Binary Search to find the student ID 67. Which algorithm is more suitable for this system? Justify your answer.	
3. A.	For the recursive function below, **explore** the base case and the recursive case, and **show** what would happen if the base case were not there. `int findMax(int a[], int n) { int m; if (n == 1) return a[0]; m = findMax(a, n - 1); if (a[n - 1] > m) return a[n - 1]; return m; }`	[CLO-3, Level-3] [Marks-2+4]
B.	**Scenario:** At the DIU Software Engineering Fest, seven food stalls were set up in a row. By evening, the first stall had sold 25 plates of fuchka, the second 47, the third 33, the fourth only 18, the fifth 64, the sixth 56, and the last stall 41. The organisers stored these sales in an array in the order of the stalls. To give a prize, they want to check whether any stall sold exactly **47** plates, using **Binary Search**. **Apply** all the steps needed to find 47 with Binary Search, and state where it is found.	
4. A.	A hospital uses a singly linked list to maintain the queue of patients waiting for medical consultation. Each patient is represented by one node. Each node occupies 8 bytes, and memory is allocated sequentially from addresses 5000, 5008, 5016, 5024, 5032, 5040, and 5048. Initially, six patients are added to the linked list in the following order: P101, P102, P103, P104, P105, P106. **Show** the following operations: a) **Sketch** the initial linked list containing all six patients in the given order. b) Three emergency patients arrive and must be inserted at specific positions: - P109 must be inserted immediately after P102. - P110 must be inserted immediately after P104. - P111 must be inserted at last. **Sketch** each updated linked list, showing the node sequence, memory addresses, and next pointers. c) If we want to prepare a registry of all the patients' ID, Name, Mobile, and address, how will the structure be written in a C program?	[CLO-3,Level-3] [Marks-6]

ourdiu.com | Shared by Aszak Ahmed Fahim

6. A web browser maintains a history of recently visited pages using a stack implemented with an array. The most recently visited page always stays on top, and when a page is removed, the most recently visited page is removed first. The maximum capacity of the stack is 5. Initially, the browser history contains the following pages: P1, P2, P3, P4, where P4 is the most recently visited page. [CLO-3, Level-3] [Marks-5]

Illustrate the following operations step by step:

a) **Draw** the initial arrangement of the browser history, clearly indicating the TOP position and array index positions.

b) Two new pages, P5 and P6, are visited and added to the browser history. **Show** the stack after each insertion.

c) Four pages are removed from the browser history one after another. **Show** the stack after each removal.

d) Attempt to remove two more pages consecutively. **Show** the stack after each operation and indicate if any stack underflow occurs.

e) **Show** the main part of a C program to POP a new page into the browser history, checking whether the stack is full before insertion.

5000