



Daffodil International University

Department of Software Engineering

Faculty of Science & Information Technology

Midterm Exam Examination, Spring 2025

Course Code: SE1131, Course Title: Data Structure

Level: 1 Term: 3 Section: A, B, C, D, E, F, G, H, I, J, K

Instructor: AB, RI, DMA, MSP, MRD, TM, KF, And SAM

Modality: Physical

One and a half hours (1:30 Hrs)

Marks: 25

Directions:

- Students need to go through the CASE STUDY shown in this exam paper.
 - Analyze and answer specific section based on your own thinking and work. Answer questions serially.
- A. Describe the advantages of a Linked List over an Array with examples. Explain how these advantages impact the performance of data structures [CLO-1, Level-1] [Marks-2]
B. Define linear and non-linear data structures and identify key differences with examples. [CLO-1, Level-1] [Marks-2]
 - The students of the 42nd batch of the Software Engineering Department decided to go on the Ruma Thanchi Circuit trek. Before the trek, they tracked the weather temperatures to ensure they are fully prepared to tackle the adventure ahead. They stored the temperatures in an unsorted array. [CLO-2, Level-2] [Marks-4]

30	28	31	27	32
----	----	----	----	----

- Insert 29 at index 2 and Describe the iteration steps.
 - Delete 31 from the array, Describe the iteration steps, and Compute its time complexity.
- A teacher has a list of unsorted student marks from a recent exam. The marks are as follows: [72, 85, 90, 45, 60] [CLO-3, Level-3] [Marks-6]
 - If you need to sort, illustrate the pseudocode of bubble sort to sort all the data in ascending order.
 - After sorting, a student named Alex wants to check if their score, 85, is on the list. For that, apply an efficient search technique.
 - Now, illustrate the time complexity of that efficient search technique.
 - A. Suppose, You need to store the names of the 21st February martyrs: "Salam", "Rafiq", "Barkat", "Jabbar", "Barkat" into a linked list. There are available memory addresses like:

1952	1966	1969	1971	2024	1111
------	------	------	------	------	------

Illustrate the following operation:

[CLO-3, Level-3] [Marks-8]

- Sketch the linked list. [2]
 - Sketch the linked list, After deleting "Rafiq" and "Jabbar". [2]
 - Again sketch the linked list, after inserting the "Shafiq" and "Sobhan" as first and last elements. [2]
 - Illustrate the line of the code for C program to create the structure of a node with two data and one pointer. [2]
- You are the chief engineer on a space mission, responsible for stacking satellite modules before launch. The rocket can only carry 5 modules at a time. The current stack of modules is Alpha, Beta, Gamma, Delta, and Epsilon. Perform the following operations and illustrate the state of the stack after each step mentioning the current top value: [CLO-3, Level-3] [Marks-3]
 - Draw the initial stack of satellite modules.
 - Show the status of the stack after Pushing "Mercury" into the stack.
 - Sketch the stack after Popping 3 modules
 - Draw the stack after Pushing "Solaris" into the stack.
 - Sketch the stack after Popping 4 modules.
 - Draw the stack after Pushing "Polaris" into the stack.

OS - mid - Spring 2025

Question 8(1)

a) Advantages of linked list over array:

1) Dynamic Search: Linked list can grow or shrink dynamically during runtime but array has fixed sizes and resizing it is tough.

2) Efficient Insertion/Deletion: No shifting is needed when inserting and deleting an element. Only the adjustment of pointers is needed. with time complexity of $O(1)$.

b) Linear vs non linear

1. Linear: A Linear data structure organizes its elements in sequential manner. where each element has a unique predecessor and successor. Example: Arrays, linked list, stack, queues.

2. Non-Linear: A non linear data structure allows elements to connect to multiple other elements, forming hierarchy or networks.

Example: trees and graph.

difference The fundamental difference lies in connectivity. Linear structure enforces one-one connection and non linear enforces many-many relationships.

Question 2:

Question 2:

a) Algorithm to insert 29 at index-2

Initial step,

LA = [30, 28, 31, 27, 32] $\rightarrow N=5$

Item = 29

k = 2

Algorithm:

1. Start

2. Set $j = N - 1 \rightarrow j = 4$

\Rightarrow Points to element 32

3. Set $N = N + 1 \rightarrow N = 6$

\Rightarrow conceptually, array now has 6 elements

4. Repeat 5 and 6 while ($j > k$)
 $= (j > 2)$

5. $LA[j+1] = LA[j]$

$\Rightarrow LA[5] = LA[4]$

$LA[5] = 32$ } example

6. $j = j - 1$

$\Rightarrow j = 3$

7. Set $LA[k] = \text{Item}$

$\Rightarrow LA[2] = 29$

8. Stop

b) Deleting 31 from the array.

1. Start

2. Set $I = 0$

3. While $I < N$ and $LA[I] \neq 31 : I = I + 1$

4. If $I = N \rightarrow$ Not found \rightarrow Stop

5. Set $J = I$

6. While $J < N - 1 :$

7. $LA[J] = LA[J + 1]$

8. $J = J + 1$

9. Set $N = N - 1$

10. Stop

Question 3:

a) Pseudocode of bubble sorting:

Let,

int array[5] = {72, 85, 90, 45, 60}, c, d, n=10;

for (c=0; c < n-1; c++) {

for (d=0; d < n-1-c; d++) {

if (array[d] > array[d+1]) {

int temp = array[d];

array[d] = array[d+1];

array[d+1] = temp;

b)

⑤ The most efficient technique - binary

Search. As the array is now sorted.

$([l+6] \text{ break})$
 $first = 0, last = 5-1, Search = 85;$

$[6] \text{ middle} = (0+4)/2$

$[l+6] \text{ while } (first \leq last) \{$

$first = [l+6] \text{ if } (array[middle] < Search) \{$

$first = middle + 1;$

$\}$

$\text{else if } (array[middle] == Search) \{$

$\text{printf}("\%d \text{ found at location } \%d \text{ m}",$

$Search, middle + 1);$

$\text{break};$

else

$last = middle - 1$

$middle = (first + last)/2$

$\}$

if (first > last)

printf("Not found");

②

Complexity:

In binary search the search space is halved in each step. After k comparisons the remaining size is $\frac{n}{2^k}$

$$\text{Set } \frac{n}{2^k} = 1 \rightarrow 2^k = n \rightarrow k = \log_2 n$$

\therefore time complexity is $= O(\log n)$

Question 4:

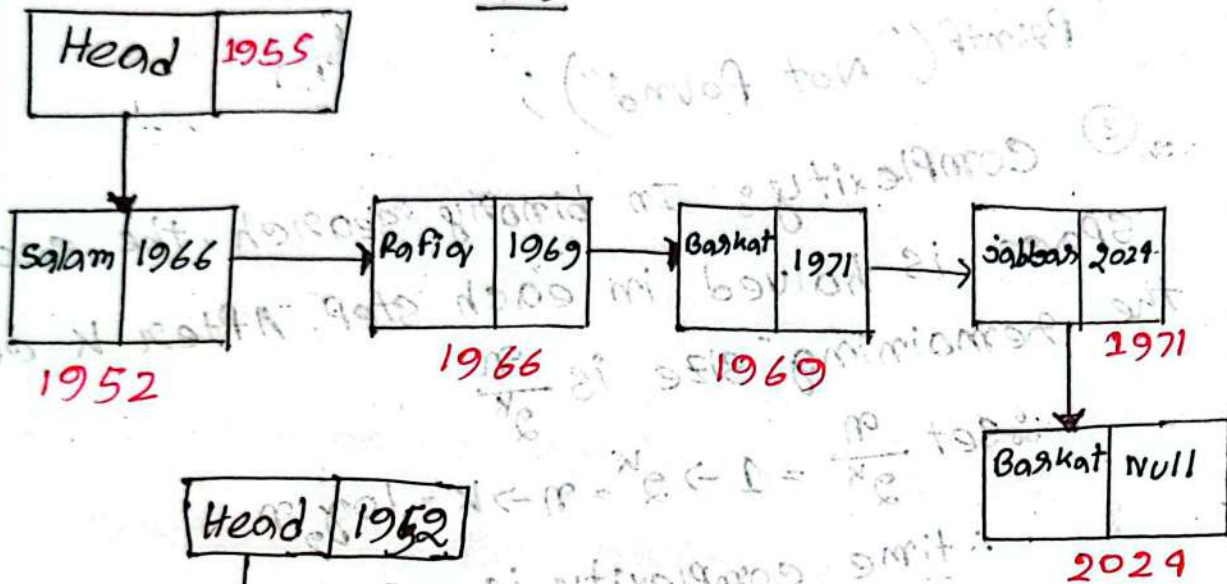
a) The linked list is:

cell nodes

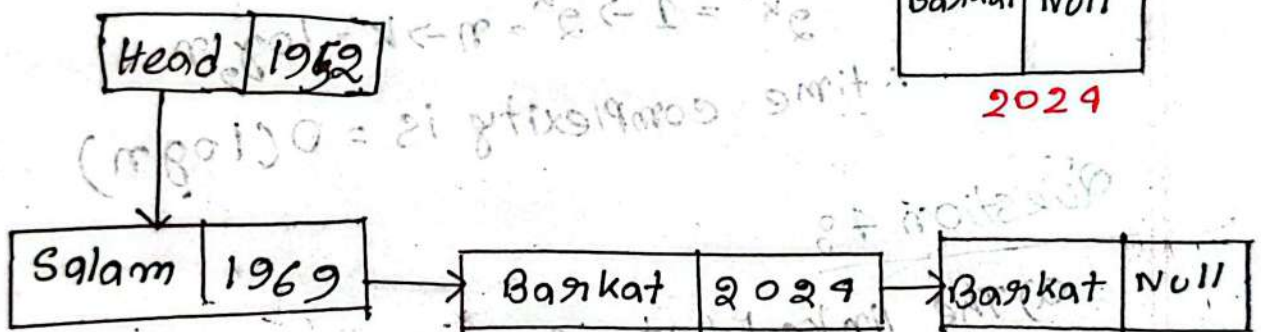
①

Question 4:

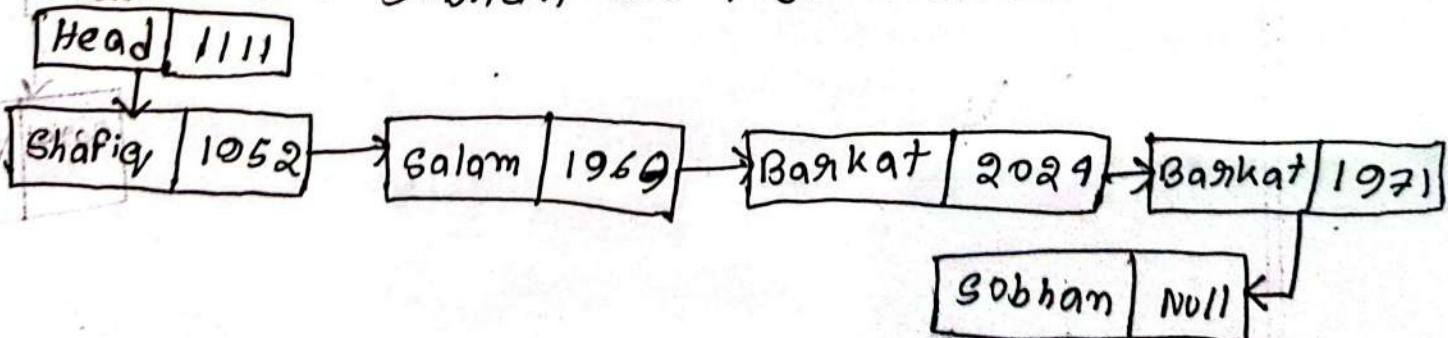
(a)



(b)



(c) As from (b) we have deleted 2 node items but their memory location is still in the Heap. So we will utilize them to insert Shafiq and Sobhan as their respected positions.



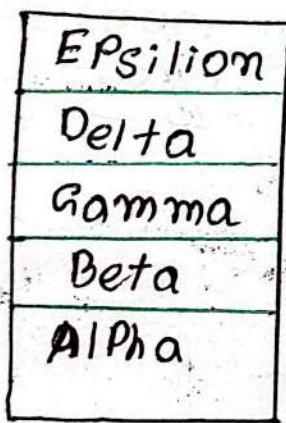
④ A node cant contain 2 data and one pointer.
this (should be a printing mistake)

2 data 1 pointer node:

```
Struct Node {  
    char name[50];  
    int Id;  
    Struct Node* Next;  
};
```

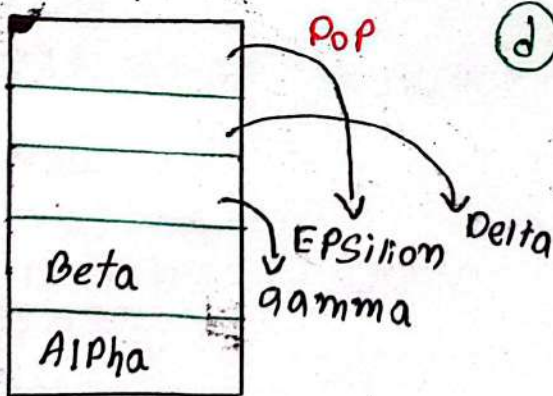
③

①

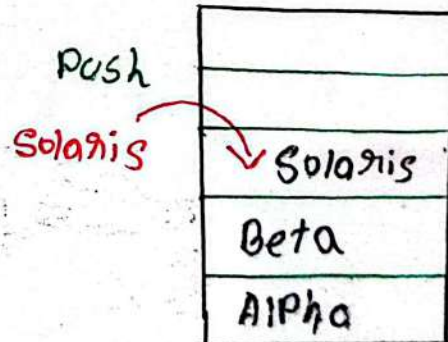


② AS the stack can only take 5 modules, so pushing Mercury wont be possible and it will cause stack overflow.

③



④



② Current stack has only 3 modules

→ Solaris, Beta, Alpha.

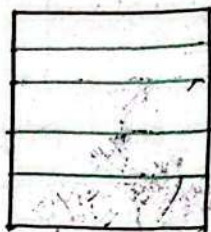
Popping 4 modules would cause underflow after 3 pops.

Pop 1 → Remove Solaris → top = Beta

Pop 2 → Remove Beta → top = Alpha

Pop 3 → Remove Alpha → top = empty

Pop 4 → Underflow — no more elements



③

Push
Solaris

