

DS

Assignment - 1

Q1. Select correct answer

- 1) Which one of the following is not the part of ADT description?

Ans: c. Both Data & operations

- 2) An ADT is defined to be a mathematical model of a user-defined type along with the collection of all _____ operations on that model

Ans: c. Primitive

- 3) Which of the following is not the part of ADT description?

Ans: c. Both of the above

- 4) Which of the following data structure is linear type?

Ans: c. Stack

- 5) What is the output of this code?

Ans: a. 123450

Q2) Choose Correct options

1) What will be the output of the program if the array begins at address 05486?

Ans: a. 65486

2) what will be the output of the program?

Ans: b. 4

3) which of the following statements are correct (about an array).

Ans: b. 1, 4

4) a. 3 and 5

5) a. p1 + 2

6) Which of the following statements are correct about 6 used in the program?

```
int num[6];  
num[6]=21;
```

Ans: B. In the first statement 6 specifies a array size, whereas in the second statement it specifies a particular element of array.

PAGE NO.	
DATE	/ /

7) What will be the output of the program?

Ans: b. 4

8) Which of the following commands will create a list?

Ans: d. All of the mentioned

9) Suppose list Example is ['h', 'e', 'l', 'l', 'o'], what is len(list Example)?

Ans: a. 5

10) Suppose list1 is [2, 33, 222, 14, 25], what is list1[-1]?

Ans: b. 25

Q3) Answer the following questions in brief.

i) What is abstract data type? Write an ADT for an array.

Ans: Abstract data type is a mathematical specification of the data, a list of operations that can be

carried out on that data.
`int [] arr = new int [1];`

- (2) Explain two-dimensional arrays with row & column major implementation. Explain address calculation in both cases with example.

Ans: In Row-Major form, all the elements of the first row are printed, then the elements of the second row and so on upto the last row. In Column-major form, all the elements of the first column are printed, then the elements of the second column and so on upto the last column.

Address calculation

Row major

For any array `int M[R][C]`; where $R = \text{no. of rows}$ & $C = \text{no. of columns}$, the location of an element $M[i][j]$ in the array can be calculated as:

Address of $M[i][j] = \text{Base}$

Address + $i * C * \text{Element size}$

+ $j * \text{Element size}$

= Base Address + $(i * C + j) * \text{Element size}$

Address Calculation

Column

For an array in $M[R][C]$, where $R =$ no. of rows & $C =$ no. of columns, the location of an element $M[i][j]$ in the array can be calculated as:

$$\begin{aligned} \text{Address of } M[i][j] &= \text{Base Address} \\ &\quad + j * R * \text{Element size} + i * \\ &\quad \text{element size} \\ &= \text{Base address} + (j * R + i) * \text{Element} \\ &\quad \text{size} \end{aligned}$$

- (3) Explain the types of Linear Data structure.

Ans: Linear Data Structure: Arrays

An array is a collection of homogeneous data elements described by a single name. Each element of an array is referenced by a subscripted variable or value, called subscript or index enclosed in parenthesis.

Linear Data Structure: Stack

A stack is one of the most imp. and useful non primitive linear

PAGE NO.	
DATE	/ /

data structure in computer science.

Linear Data Structure: Queue

A queue is logically a first in first out (FIFO) linear data structure.

Linear Data Structure: Linked List

Linked list is a dynamic data structure which is an ordered collection of homogeneous data elements called nodes, in which each element contains two parts: data or info and one or more links.

- 4) Explain the types of nonlinear data structure)

Ans: Graph:

A graph can be represented as $G = (V, E)$. Where V is a finite non-empty set of vertices & E is a set of pairs of vertices called edges. Each edge e in E is identified with a

unique pair (a, b) of nodes in V
denoted by $\sigma = [a, b]$

Tree:

Tree can be defined as a ~~finite~~ set of one or more data items such that:

1. There is a special node called the root of the tree.
2. Remaining nodes are partitioned into no. of mutually exclusive subsets each of which is itself a tree, are called subtrees.
- 3) Explain difference between primitive and non-primitive data structures.

Ans : Primitive Data Structure	Non-Primitive Data Structure
--------------------------------	------------------------------

- A primitive data structure is generally a basic structure that is usually built into the language, such as an integer, a float.
- A non-primitive data structure is built out of primitive data structures linked together in meaningful ways, such as a linked-list, binary search tree, AVL tree, graph.