

# \* Data Structure Lab (DSL):- Practical Number - 2 (Group - A)

Name:- Kaustubh Shrikant Kabra.

Class:- Second Year Engineering

Div:- A

Roll Number:-

Batch:-

Department:- Computer Department

College:- AISSMS's IOIT.

Title:-

Write a python program to store marks scored in a subject by students in the class.

Aim:-

Write a python program to store marks scored in subject 'Fundamental of Data Structure' by N students in the class. Write function to compute following:-

- 1) The average score of class
- 2) Highest and lowest score of class.
- 3) Count of students who were absent for the test.
- 4) Display marks with highest frequency.

Objective:-

- 1) To study the concept of list in python.
- 2) To understand various operations on lists.

Theory:-

List-

List is a versatile datatype available in python. It is a sequence



in which elements are written as a list of comma separated between square brackets, the key feature of a list is that it can have elements that belong to different datatypes.

Syntax  $\rightarrow$  `<list_variable> = [val1, val2, ...]`

Examples:-

1) `A = [1, 2, 3, 4, 5]`

2) `B = [1, 3, 5, 7, 'A', 'B', 'C', "Hello"]`

Accessing Lists -

Similar to strings, lists can also be sliced and concatenated. To access values in lists, square brackets are used to slice along with index.

Example:-

`A = [1, 2, 3, 4, 5, 6, 7, 8]`

`print(A[0])`

// 1

`print(A[3])`

// 4

`print(A[1:4])`

// 2 3 4

Updating Lists -

Once created, values in a list can be easily updated using the index and the assignment operator.

Example:-

`A = [1, 2, 3, 4]`

`A[2] = 100`

`print A`

// 1 2 100 4



## Nested Lists -

Nested list means a list within another list. List can have element of different datatypes which can include even a list.

Example:-

```
list1 = [1, 'a', "abc", [2, 3, 5, 7], 8.9]
```

Algorithm:-

Step 1 - Start

Step 2 - Read number of students from the user (N).

Step 3 - Accept marks in FDS for the total number of students.

Step 4 - Append the marks into a list.

Step 5 - Write a function to calculate average marks of students.

Step 6 - Display the average marks of the class.

Step 7 - Write function to calculate highest and lowest marks in class.

Step 8 - Display the highest and lowest marks.

Step 9 - Write a function to calculate number of absent student.

Step 10 - Display number of absent students.



Step11 - Write a function to calculate the frequency of the entered marks.

Step12 - Display the marks with their corresponding frequencies.

Step13 - Display marks with highest frequency along with its frequency.

Step14 - Stop.

Analysis:-

The time complexity for `average()`, `highestMarks()`, `lowestMarks()`, `absentstudents()` and `frequency` is  $O(n)$ .

Conclusion:-

Hence, we have used list to store marks and have performed various operations on it.