Experiment No-02 (Group A)

Title:- Write a python program to store marks for N students

Objectives:- To understand the use functions for N students record.

Problem Statement:-

Write a Python program to store marks scored in subject “Fundamental of DataStructure” by N students in the class. Write functions to compute following:

a) The average score of class
b) Highest score and lowest score of class
c) Count of students who were absent for the test
d) Display mark with highest frequency

Outcome: Resulting average, highest and lowest marks operation.

Software and Hardware requirements:-

1. **Operating system:** Linux- Ubuntu 16.04 to 17.10, or Windows 7 to 10,
2. **RAM-** 2GB RAM (4GB preferable)
3. You have to install **Python3** or higher version

Theory- Input & Output in python Programming

Input means the data entered by the user of the program. In python, the input( ) function is used to accept an input from user

Syntax: varaiable= input( ) OR variable = input (“ enter data”)

```
e.g. >>>input( )
Hello Python
Hello Python
>>>x = input(“Enter your name”)
Enter your name ishwar bharambe
>>>print(x)
Ishwar bharambe
```

Output means the data comes from computer processing. In python programming the print( ) function display the input value on screen

Full syntax of print( ) is **print(*objects, sep= „ „ , end=‟\n‟)**

where *object- object to be printed, * indicates that there may be more than one object

sep – object are separated by sep. Default value is „ ‟

end – end is printed at last

```
e.g. >>>name = „ishwar‟
>>>age = 31
>>>print(name,age,sep=‟\n‟,end=‟\n\n\n‟)
>>>print(name,age) # ishwar 31
>>>print(„my name is‟,name,‟and age is‟,age) # my name is ishwar and age is 31
>>> print(„my name is %s and age is %d‟%(name,age)) # my name is ishwar and age is 31
>>> print(„my name is {} and age is {}‟.format(name,age) # my name is ishwar and age is
31
```

Array in python:-

Array in python is implemented by using list.

A List is a collection of ordered python objects separated by commas and changeable.

In Python List are written with square brackets.

To create a List in Python, place all the elements in a [ ] square brackets, separated by commas.

A List can have heterogeneous data items, a tuple can have string and list as data items as well but list are commonly used as homogenous objects.

e.g. >>>list1 = [„apple‟,‟banana‟, „cherry‟,‟kiwi‟,‟orange‟]

>>>print(list1)

Output:- [„apple‟,‟banana‟, „cherry‟,‟kiwi‟,‟orange‟]

if..else statement:-

The if..else statement evaluates test expression and will execute body of if only when test condition is True.

If the condition is False, body of else is executed. Indentation is used to separate the blocks.

Syntax:

```
if (condition):
        # Executes this block if
        # condition is true
else:
        # Executes this block if
        # condition is false
```

if...elif...else Statement

The elif is short for else if. It allows us to check for multiple expressions.

If the condition for if is False, it checks the condition of the next elif block and so on.

If all the conditions are False, body of else is executed.

Only one block among the several if...elif...else blocks is executed according to the condition.

The if block can have only one else block. But it can have multiple elif blocks.

Syntax of if...elif...else

```
        if test expression:
                Body of if
        elif test expression:
                Body of elif
        elif test expression:
                Body of elif
        ……….
        else:
                Body of else
```

while loop:-

In while loop, test expression is checked first. The body of the loop is entered only if the test_expression evaluates to True. After one iteration, the test expression is checked again. This process continues until the test_expression evaluates to False.

Syntax of while loop:-

```
while   test_expression:
                Body of while
```

for loop:-

In Python, there is no C style for loop, i.e., for (i=0; i<n; i++). There is “for in” loop which is similar to for each loop in other languages.

The for loop in Python is used to iterate over a sequence (list, tuple, string) or other iterable objects. Iterating over a sequence is called traversal.

Syntax of for Loop

```
for val in sequence:
        Body of for
```

Here, val is the variable that takes the value of the item inside the sequence on each iteration.

Loop continues until we reach the last item in the sequence. The body of for loop is separated from the rest of the code using indentation. It can be used to iterate over iterators and a range.

Function definition:-

In python, we can use **def** keyword to define the function.

The function block is started with the colon (:) and all the same level block statements remain at the same indentation.

A function can accept any number of parameters that must be the same in the definition and function calling. The syntax to define a function in python is given below.

```
def function_name(parameters) :
        """docstring"""
        statement(s)
        return <expression>
```

Example:-

```
def Greeting( ):
      """This function greets to the person passed in as  parameter"""
      print("Hello, Good morning!")
```

Above shown is a function definition which consists of following components.

1. Keyword def marks the start of function header.

2. A function name to uniquely identify it. Function naming follows the same rules of writing identifiers in Python.

3. Parameters (arguments) through which we pass values to a function. They are optional.

4. A colon (:) to mark the end of function header.

5. Optional documentation string (docstring) to describe what the function does.

6. One or more valid python statements that make up the function body. Statements must have same indentation level (usually 4 spaces).

7. An optional return statement to return a value from the function.

Function calling:-

In python, a function must be defined before the function calling otherwise the python interpreter gives an error.

Once the function is defined, we can call it from another function or the python prompt.

To call the function, use the function name followed by the parentheses.

Example of calling above function

```
>>>Greeting( )
```

Example of Defining and Calling function in python:-

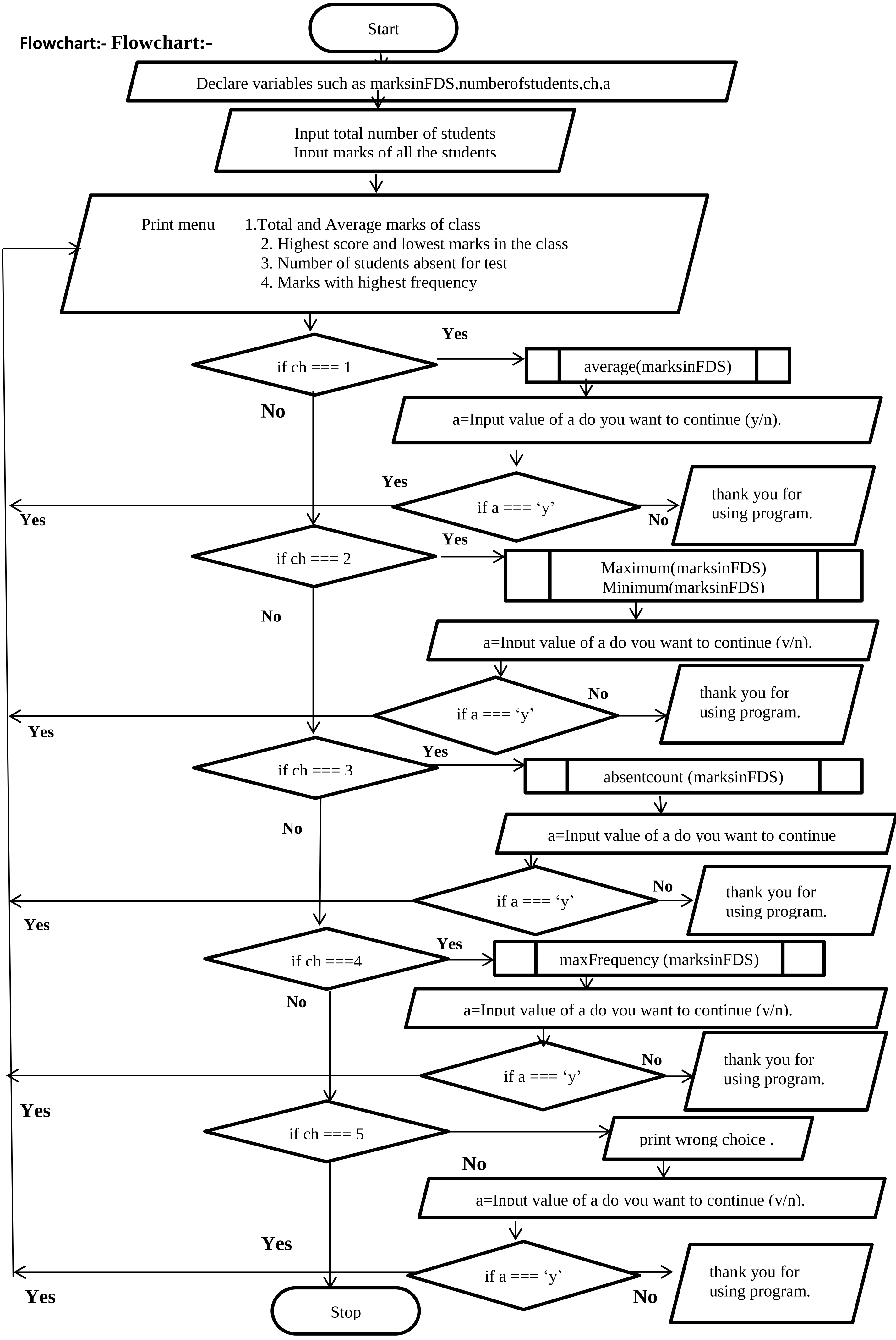
```
def Greeting( ):
        """This function greets to the person passed in as parameter"""
          print("Hello, Good morning!")
 Greeting( )  # Calling function
```

#Output:- Hello Good morning!

Algorithm:-

```
1) Start.
2) Declare variables such as marksinFDS,numberofstudents,ch,a
3) Input total number of students
4) Input marks of all the students
5) Print menu    1.Total and Average marks of class
                 2. Highest score and lowest marks in the class
                 3. Number of students absent for test
                 4. Marks with highest frequency
                 5.Exit.
                     ch=Input Enter your choice(from 1 to 5)
6)   If ch==1 then call function average()
         Input value of a do you want to continue (y/n).
         If a=='y' then go to step 5.
             else,thank you for using program.
7) else if ch==2,then call function Maximum( ) and Minimum( )
          Input value of a do you want to continue (y/n).
          If a=='y' then go to step 5.
           else ,thank you for using program.
8) else if ch==3 then call  function absentcount( )
          Input  value of a,  do you want to continue (y/n).
         If a=='y' then go to step 5.
        else, thank you for using program.
9) else if ch==4 then call function maxFrequency( )
          Input value of a , do you want to continue (y/n).
           If a=='y' ,then go to step 5.
           else,thank you for using program.
10) else if ch==5 then ,go to step 11.
           else,print wrong choice .
           Input value of a, do you want to continue (y/n)
               if a=='y' then go to step 5.
              else,thank you for using program.
11) Stop.
```

Flowchart:- Flowchart:-


Start
Declare variables such as marksinFDS,numberofstudents,ch,a
Input total number of students
Input marks of all the students
Print menu
1.Total and Average marks of class
2. Highest score and lowest marks in the class
3. Number of students absent for test
4. Marks with highest frequency
if ch === 1
Yes
No
average(marksinFDS)
a=Input value of a do you want to continue (y/n).
Yes
if a === 'y'
Yes
No
thank you for using program.
if ch === 2
Yes
No
Maximum(marksinFDS)
Minimum(marksinFDS)
a=Input value of a do you want to continue (y/n).
if a === 'y'
No
Yes
thank you for using program.
if ch === 3
Yes
No
absentcount (marksinFDS)
a=Input value of a do you want to continue
if a === 'y'
No
Yes
thank you for using program.
if ch ===4
Yes
No
maxFrequency (marksinFDS)
a=Input value of a do you want to continue (y/n).
if a === 'y'
No
Yes
thank you for using program.
if ch === 5
print wrong choice .
No
a=Input value of a do you want to continue (y/n).
Yes
if a === 'y'
Yes
No
thank you for using program.
Stop

Conclusion:

By this way, we can store the marks of N students sucessfully.