**CSA0851 – PYTHON PROGRAMMING FOR SOCIAL APPS**

**MODEL EXAMINATION**

**EASY**

1. Write a python program to check whether a given year is a leap year or not.

Sample Input:

Enter Year: 1947

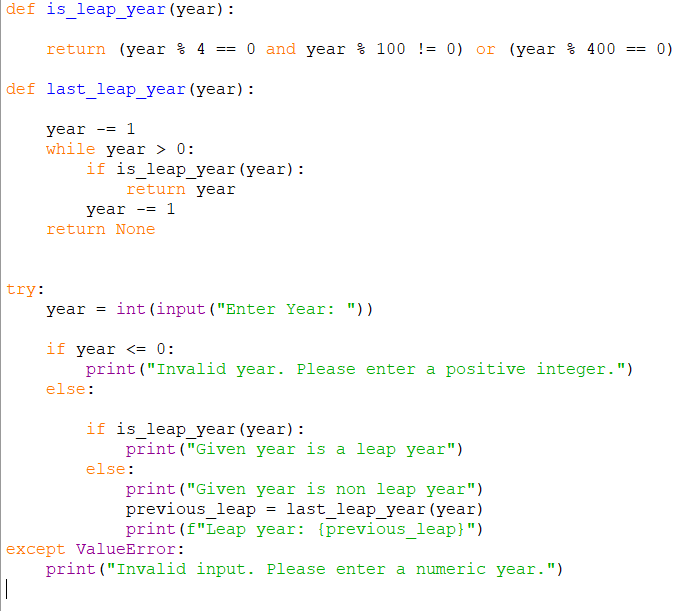
Sample Output:

Given year is non leap year

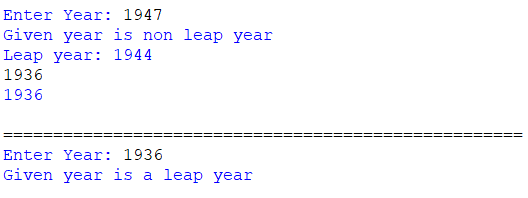
Leap year: 1944

1. 1947
2. 1936
3. 0
4. 2000
5. -1428

**PROGRAM**



**OUTPUT**



1. Write a python program to find the Mean, Median, Mode of given numbers

Sample Input:

Array of elements: {16, 18, 27, 16, 23, 21, 19}

Sample Output:

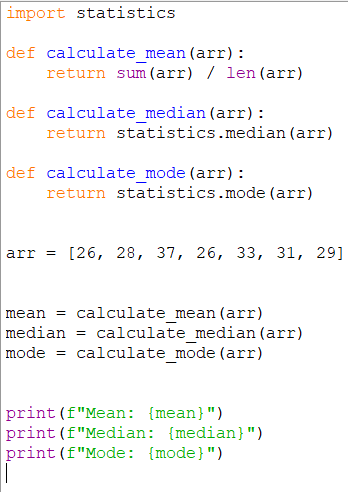
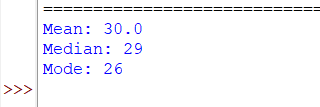
Mean: 20

Median: 19

Mode: 16

1. Array of elements: {26, 28, 37, 26, 33, 31, 29}
2. Array of elements: {1.6, 1.8, 2.7, 1.6, 2.3, 2.1, 1.9}
3. Array of elements: {0, 160, 180, 270, 160, 230, 210, 190, 0}
4. Array of elements: {200, 180, 180, 270, 160, 270, 270, 190, 200}
5. Array of elements: {100, 100, 100, 100, 100, 100, 100, 100, 100}

**PROGRAM OUTPUT**

**** ****

1. Find the Mth maximum number and Nth minimum number in an array and find the sum and difference of it

Sample Input:

Array of elements: {14, 16, 87, 36, 25, 89, 34}

M=1

N=3

Sample Output:

1st Maximum Number = 89

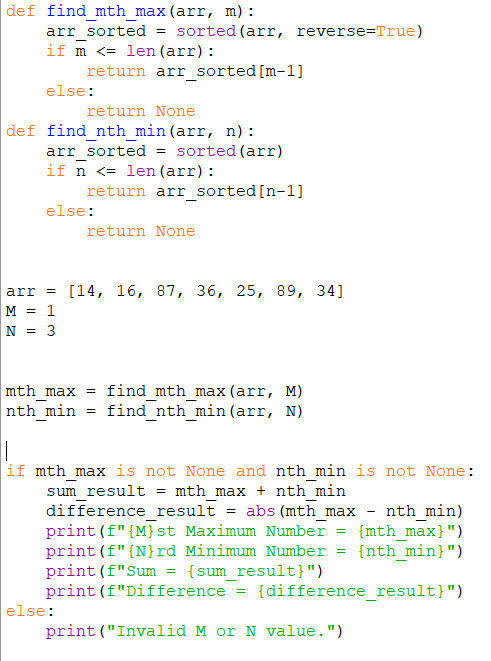
3rd Minimum Number = 25

Sum = 114

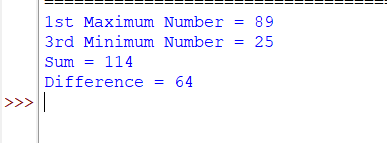
Difference = 64

1. {16, 16, 16, 16, 16}, M=0, N=1
2. {0,0,0,0}, M=1, N=2
3. {-12, -78, -35, -42, -85}, M=3, N=3
4. {15, 19, 34, 56, 12}, M=6, N=3
5. {85, 45, 65, 75, 95}, M=5, N=7

**PROGRAM**

****

**OUTPUT**

****

1. Write a program for interesting an element to the array at the specified index

Sample Input:

Enter the number of elements: 5

Enter the elements:

47 34 21 89 12

Element to be interested 100

At position: 4

Output:

12 21 34 100 47 89

1. Number of elements: 4

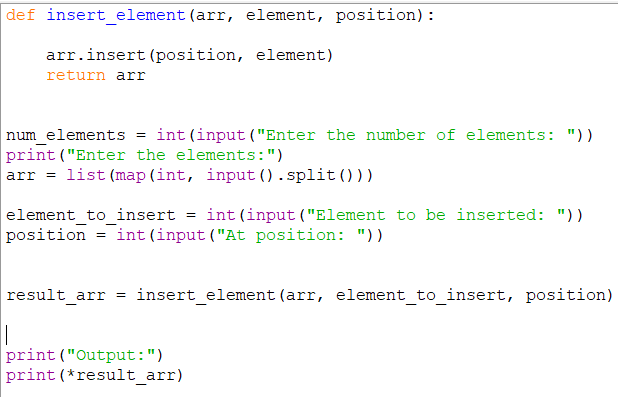
Enter the elements:

32 41 25 86 90

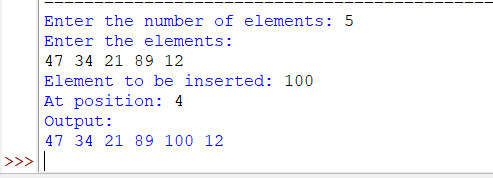
Elements to be inserted 50

At position: 2

**PROGRAM**

****

**OUTPUT**

****

1. Write a python program to read the numbers until-1 is encountered. Find the average of positive numbers and negative numbers entered by user.

Sample Input;

Enter -1 to exit

Enter the number: 7

Enter the number: -2

Enter the number: 9

Enter the number: -8

Enter the number: -6

Enter the number: -4

Enter the number: 10

Enter the number: -1

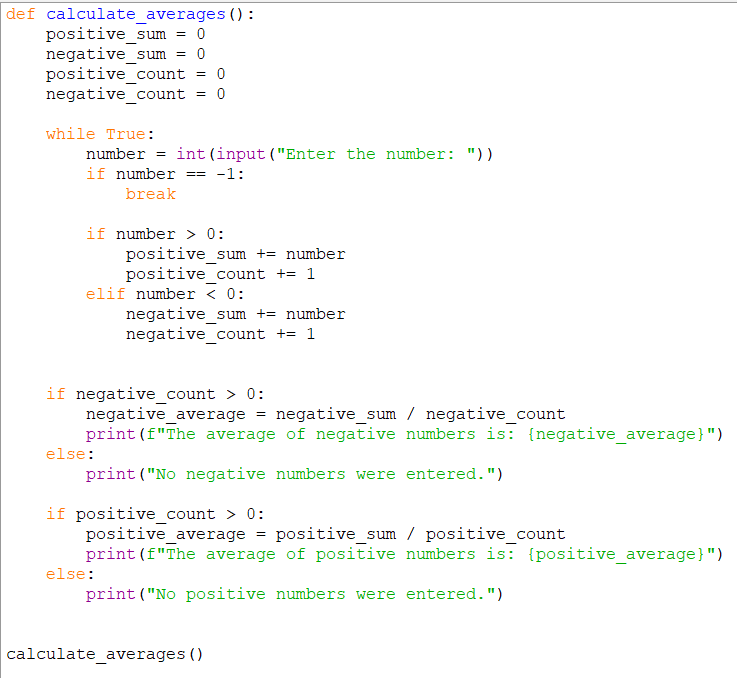
Sample Output:

The average of negative numbers is: -5.0

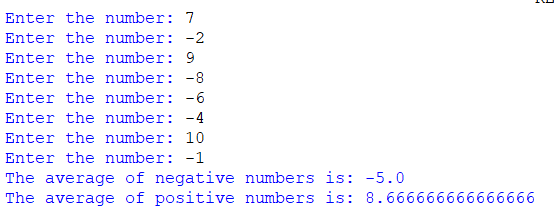
The average of positive number is: 8.66666667

1. -1, 43, -87, -29, 1, -9
2. 73, 7, -6, 2, 10, 28, -1
3. -5, -9, -46, 2, 5, 0
4. 9, 11, -5, 6, 0, -1
5. -1, -1, -1, -1, -1

**PROGRAM**

****

**OUTPUT**

****

**MEDIUM**

1. In daily share trading a buyer buys shares in the morning and sells them on the same day. If the trader is allowed to make at most two transactions in a day, whereas the second transaction can only start after the first one is complete (Buy 🡪 cell 🡪 Buy 🡪 cell). Given stock prices throughout the day, find out the maximum profit that a short trader could have made.
2. Input: prices = [7, 1, 5, 3, 6, 4]

Output: 7

1. Input: prices = [7, 6, 4, 3, 1]

Output: 0

1. Input: [10, 22, 5, 75, 65, 80]

Output: 87

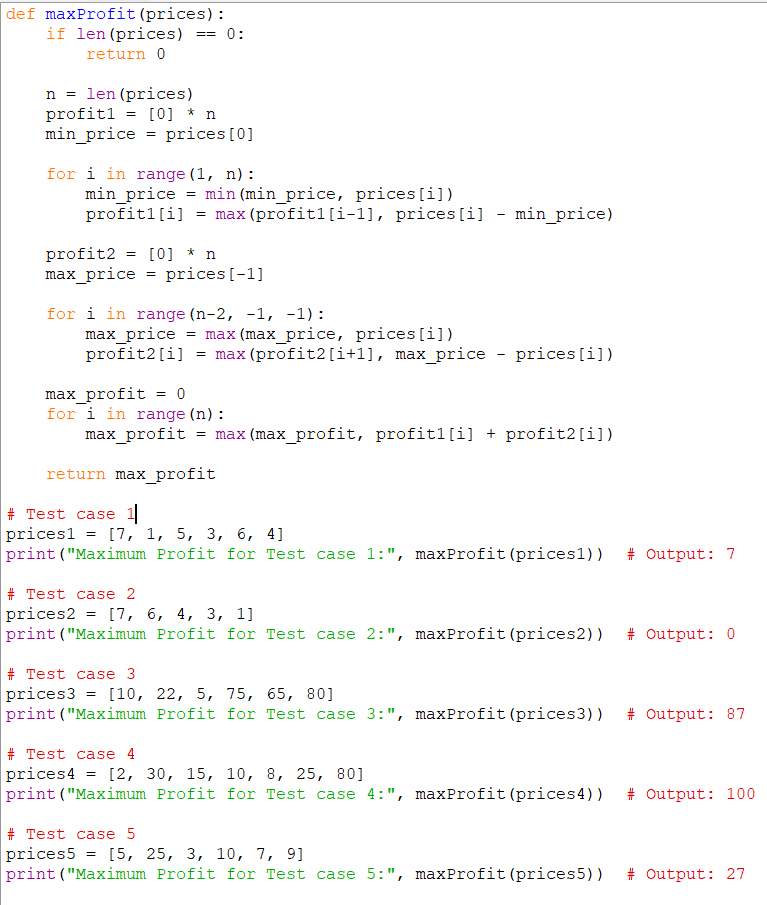
1. Input: [2, 30, 15, 10, 8, 25, 80]

Output: 100

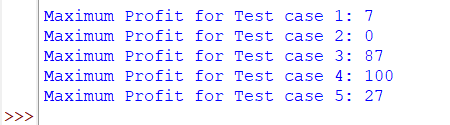
1. Input: [5, 25, 3, 10, 7, 9]

Output: 27

**PROGRAM**

****

**OUTPUT**

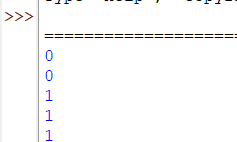
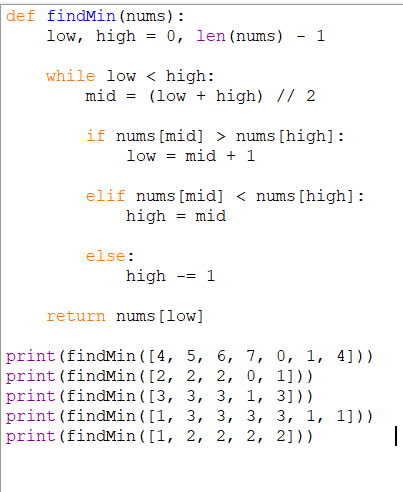


1. Suppose an array of length n sorted in ascending order is rotated between 1 and n times. For example, the array numbers = [0,1,4,4,5,6,7] might become: [4,5,6,7,0,1,4] if it was rotated 4 times. [0,1,4,4,5,6,7] it was rotated 7 times.

Notice that rotating an array [a[0], a[1], a[2],….a[a-1]] I time results in the array [a[n-1], a[0], a[1], a[2],…., a[n-2]].

Given the sorted rotated array numbers that may contain duplicates return the minimum element of this array.

**PROGRAM OUTPUT**

****

**HARD**

1. Write a Python program to find whether it contains an addictive sequence or not. The additive sequence is a sequence of numbers where the sum of the first two numbers is equal is equal to the third one.

Sample additive sequence: 6, 6, 12, 18, 30

In the above sequence 6+6=12, 6+12=18, 12+18=30….

Also, you can split a number into one or more digits to create an additive sequence.

Sample additive sequence: 66121830

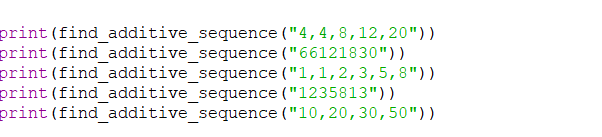
In the above sequence 6+6=12, 6+12 = 18, 12+18=30….

Input: “4, 4, 8, 12, 20”

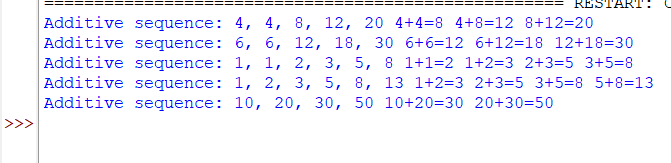
Output: Additive sequence 4+4=8, 4+8=12, 8+12=20.

**PROGRAM**

****

****

**OUTPUT**

****