

Expt. No. 7Page No. 13Expt. Name. Compute:  $s = (f(x_1) + f(x_2) + \dots + f(x_n)) \text{ Modulo } z$  Date: 08-01-2022Aim:

To write a python program by considering a function  $f(x) = x^3$ .  
 Input is 'N' list. Each list contains 'M' elements. From the list, find the maximum element. Compute:  $s = (f(x_1) + f(x_2) + f(x_3) + \dots + f(x_n)) \text{ Modulo } z$

Algorithm:

- step 1: Start
- step 2: Get the input value of m & n from the user
- step 3: initialize a empty list l and mx
- step 4: using double for loop for N and M
  - step 4.1: Get the input & append in mx
- step 5: initialize a empty list l = []
- step 6: Get the input value of z from the user
- step 7: initialize s is equal to zero
- step 8: for loop in mx
  - step 8.1: increment s by increasing the power of 3
- step 9: print s/z
- step 10: Stop

Program:

```
def f(x):
    return x**3
N = int(input('Enter N:'))
M = int(input('Enter M:'))
l = []
mx = []
for i in range(N):
```

Expt. No. \_\_\_\_\_

Page No. 14

Expt. Name. \_\_\_\_\_

Date : \_\_\_\_\_

```
for j in range(M):  
    l.append(int(input("enter elements: ")))  
mx.append(max(l))  
l = []  
z = int(input("Enter z: "))  
s = 0  
for i in mx:  
    3  
    st = f(i)  
print(s%z)
```

### Result:

The above code is executed successfully and the output is attached.



Expt. No. 8

Page No. 15

Expt. Name. Validate the Credit Numbers based on the following

Date: 08-01-2022

Aim:

To write a python program by validating the Credit numbers based on the following condition: Begins with 4, 5 or 6, contain exactly 16 digits, contains only number (0 to 9). For every 4 digits a hyphen (-) may be included (not mandatory). No other special character permitted. Must not have 4 or more executive same digits.

Algorithm

- step 1: start
- step 2: import the library re and itertools
- step 3: ~~if~~ initialize the text
- step 4: print the length of the text
- step 5: initialize the l to k, sum(1 for i in g) for k, g in itertools.groupby(~~text~~)
- step 6: check the condition, if yes print 'passed' else ~~'failed'~~ "False"
- step 7: Stop

Program:

```
import re
import itertools
text = "5133-3387-8912-3456"
print(len(text))
l = [(k, sum(1 for i in g)) for k, g in itertools.groupby(text)]
if re.search(r'^[456]?', text) and len(text) == 16 and re.search(r'[10]', text)
and all(v <= 3 for k, v in l) and bool(re.search(r'[10]', text)) is False
and bool(re.search(r'[a-z]', text)) is False or (bool(re.search(r'[-]', text))
is True and len(text) == 19):
```



Expt. No. \_\_\_\_\_

Page No. 16

Expt. Name. \_\_\_\_\_

Date : \_\_\_\_\_

```
print("it passed")
```

```
else:
```

```
print("Failed False")
```

Result:

The above code is executed successfully and the output is attached.