

```
In [1]: #Wap that takes a positive integers and returns the sum of the cube of all the posit
a=int(input("Enter te Specified No.:- "))
n=int(input("No. of time you want to enter integer:- "))
s=0
for x in range(0,n):
    m=int(input("Enter the Positive Integer :- "))
    if(m<a):
        s+=m*m*m
print("The sum is:- ",s)
```

```
Enter te Specified No.:- 5
No. of time you want to enter integer:- 3
Enter the Positive Integer :- 1
Enter the Positive Integer :- 2
Enter the Positive Integer :- 6
The sum is:- 9
```

```
In [2]: #Wap that takes a sequence of numbers and determine whether all the numbers are diff
lst=[]
n=int(input("enter size"))
for i in range(0,n):
    x=int(input())
    lst.append(x)

flag=True
for i in range(0,n):
    for j in range(i+1,n):
        if(lst[i]==lst[j]):
            flag=False
            break
if(flag):
    print("UNIQUE")
else:
    print("NOT UNIQUE")
```

```
enter size5
1
2
2
3
4
NOT UNIQUE
```

```
In [16]: #Wap to remove and print every third number from a list of numbers until the list be
l=[]
size=int(input("Enter list size"))
for i in range(size):
    l.append(int(input()))

print('removal operation:')
i=2
while len(l)>2:
    if i>=len(l):
        i=i-len(l)
    print(l.pop(i))
    i=i+2

if i==4:
    print(l.pop(0))
else:
    print(l.pop())

print(l.pop())
```

```

Enter list size5
1
2
3
4
5
removal operation:
3
1
5
2
4

```

```

In [5]: #Implement Binary Search using List data structure.
def binary_search(arr,low,high,x):
    if high>=low:
        mid=(high+low)//2
        if arr[mid]==x:
            return mid
        elif arr[mid]>x:
            return binary_search(arr,low,arr[mid]-1,x)
        else:
            return binary_search(arr,mid+1,high,x)
    else:
        return -1
arr= [2,3,4,10,40]
x=10
result=binary_search(arr,0,len(arr)-1,x)
if result!=-1:
    print("Element is present at index",str(result))
else:
    print("Element is not present in array")

```

Element is present at index 3

```

In [6]: #Use List Data Structure to multiply two matrices.Enter elements in a tuple- find Ma
R = int(input("Enter the number of rows:"))
C = int(input("Enter the number of columns:"))

X= []
print("Enter the values")

for i in range(R):
    temp1=[]
    for j in range(C):
        temp1.append(int(input()))
    X.append(temp1)

Row = int(input("Enter the number of rows:"))
Col = int(input("Enter the number of columns:"))

Y = []
print("Enter the values")

for i in range(Row):
    temp2 =[]
    for j in range(Col):
        temp2.append(int(input()))
    Y.append(temp2)

#L=[0 for i in range(Col)]

```

```
#result=[l for i in range(R)]
result = [[0 for i in range(Col)] for j in range(R)]

for i in range(len(X)):
    for j in range(len(Y[0])):
        for k in range(len(Y)):
            result[i][j] += X[i][k] * Y[k][j]

for r in result:
    print(r)

#2nd part
def tuplemax(tup,K):
    tup = list(tup)
    temp = sorted(tup)
    result = tuple(temp[:K] + temp[-K:])

    print("Max and Min K elements : ",result)

l=[]
size=int(input("Enter list size"))
for i in range(size):
    l.append(int(input()))
tup=tuple(l)

#tup = (13, 10, 23, 2, 5, 6, 12, 7, 1, 8)
K = int(input("enter vales you want"))
print("The original tuple: ", tup)
tuplemax(tup,K)
```

In [24]:

```
#Write a function matchparenthesis(s) that uses string s to see if the starting and
def matchparenthesis(s):
    ope = []
    clo = []
    for i in range(0,len(s)):
        l = s[i]
        if l == "(":
            ope = ope+"("
        else:
            if l == ")":
                clo = clo+")"
            else:
                return(ope, clo)
    if len(ope)==len(clo):
        print(True)
    else:
        print(False)
```

In [15]:

```
#Wap to read a string text and count the numbers of characters in the string and sto
def char_frequency(str1):
    dict = {}
    for n in str1:
        keys = dict.keys()
        if n in keys:
            dict[n] += 1
        else:
            dict[n] = 1
    return dict
print(char_frequency('google.com'))

{'g': 2, 'o': 3, 'l': 1, 'e': 1, '.': 1, 'c': 1, 'm': 1}
```

In []: