



Python Programming - 2301CS404

Lab - 7

Roll No : 24010101034

Name : Prince S. Chandpa

01) WAP to find sum of tuple elements.

Sample Input:

```
T = (10, 20, 30, 40)
```

Sample Output:

```
Sum of tuple elements = 100
```

```
In [1]: t1 = tuple(map(int, input('Enter Tuple: ').split()))
ans = 0
for i in t1:
    ans += i
print(f"Sum of tuple elements = {ans}")
```

```
Sum of tuple elements = 100
```

02) WAP to find Maximum and Minimum K elements in a given tuple.

Sample Input:

```
T = (7, 2, 9, 4, 1, 5)
K = 2
```

Sample Output:

```
Minimum 2 elements: (1, 2)
Maximum 2 elements: (9, 7)
```

```
In [16]: t2 = (*(int(i) for i in input('Enter Tuple element : ').split()),)
k = int(input('Enter K : '))

new_tuple = list(t2)

new_tuple.sort()

t2 = tuple(new_tuple)

min_tuple = t2[:k]
max_tuple = t2[-1:-k-1:-1]

print(f'Minimum {k} elements: {min_tuple}')
print(f'Maximum {k} elements: {max_tuple}')
```

```
Minimum 2 elements: (1, 2)
Maximum 2 elements: (9, 7)
```

03) WAP to find tuples which have all elements divisible by K from a list of tuples.

Sample Input:

```
tuple_list = [(2, 4, 6), (3, 6, 9), (4, 8, 12), (5, 10,
15)]
K = 2
```

Sample Output:

```
Tuples divisible by 2 : [(2, 4, 6), (4, 8, 12)]
```

```
In [20]: tuple_list = []
ans_list = []
number_of_tuple = int(input('Enter Number of element in Tuple List: '))

for i in range(number_of_tuple):
    t = (*(int(i) for i in input('Enter Tuple element : ').split()),)
    tuple_list.append(t)

k = int(input('Enter K : '))tuple_list

for t in tuple_list:
    for i in t:
        if i%k != 0:
            break
    else:
        ans_list.append(t)

print(f'Tuples divisible by {k} : {ans_list}')
```

```
Tuples divisible by 2 : [(2, 4, 6), (4, 8, 12)]
```

04) WAP to create a list of tuples from given list having number and its cube in each tuple.

Sample Input:

```
numbers = [1, 2, 3, 4, 5]
```

Sample Output:

```
List of tuples with number and its cube: [(1, 1), (2, 8),  
(3, 27), (4, 64), (5, 125)]
```

```
In [5]: numbers = [int(i) for i in input('Enter element of List').split()]  
  
numbers_with_cube = []  
  
for i in numbers:  
    t = i, i**3  
    numbers_with_cube.append(t)  
  
print(f'List of tuples with number and its cube: {numbers_with_cube}')
```

```
List of tuples with number and its cube: [(1, 1), (2, 8), (3, 27), (4, 64), (5, 125)]
```

05) WAP to find tuples with all positive elements from the given list of tuples.

Sample Input:

```
tuple_list = [(1, 2, 3), (-1, 2, 3), (4, 5, 6), (0, 1, 2),  
(-3, -4, -5)]
```

Sample Output:

```
Tuples with all positive elements: [(1, 2, 3), (4, 5, 6)]
```

```
In [26]: tuple_list = []  
ans_list = []  
number_of_tuple = int(input('Enter Number of element in Tuple List: '))  
  
for i in range(number_of_tuple):  
    t = (*[int(i) for i in input('Enter Tuple element : ').split()])  
    tuple_list.append(t)  
  
for t in tuple_list:  
    for i in t:  
        if i <= 0:  
            break  
    else:  
        ans_list.append(t)
```

```
print(f'Tuples with all positive elements : {ans_list}')
```

Tuples with all positive elements : [(1, 2, 3), (4, 5, 6)]

06) WAP to remove tuples of length K.

Sample Input:

```
tuple_list = [(1, 2), (3, 4, 5), (6,), (7, 8, 9), (10, 11)]
K = 2
```

Sample Output:

List after removing tuples of length 2 : [(3, 4, 5), (6,), (7, 8, 9)]

```
In [28]: tuple_list = []

number_of_tuple = int(input('Enter Number of element in Tuple List: '))

for i in range(number_of_tuple):
    t = (*(int(i) for i in input('Enter Tuple element : ').split()),)
    tuple_list.append(t)

k = int(input('Enter K : '))

for t in tuple_list:
    if len(t) == k:
        tuple_list.remove(t)

print(f'List after removing tuples of length {k} : {tuple_list}')
```

List after removing tuples of length 2 : [(3, 4, 5), (6,), (7, 8, 9)]

07) WAP to remove all occurrences of a given element from a tuple.

Sample Input:

```
T = (3, 5, 3, 7, 3, 9)
x = 3
```

Sample Output:

(5, 7, 9)

```
In [32]: t = tuple(map(int, input('Enter Tuple:').split()))
x = int(input('Enter x: '))

new_list = list(t)

for i in new_list:
```

```

if i == x:
    new_list.remove(x)

print(tuple(i for i in new_list))

```

(5, 7, 9)

08) WAP to remove duplicates from tuple.**Sample Input:**

T = (3, 5, 3, 7, 3, 9)

Sample Output:

(3, 5, 7, 9)

```

In [40]: t = tuple(map(int, input('Enter Tuple:').split()))

new_list = []

for i in t:
    if i not in new_list:
        new_list.append(i)

print(tuple(i for i in new_list))

```

(3, 5, 7, 9)

09) WAP to multiply adjacent elements of a tuple and print that resultant tuple.**Sample Input:**

t = (2, 3, 4, 5)

Sample Output:

Resultant tuple after multiplying adjacent elements: (6,
12, 20)

```

In [4]: t = tuple(map(int, input('Enter Tuple:').split()))

new_list = list(t)
ans_list = []
for i in range(len(new_list) - 1):
    ans_list.append(new_list[i]*new_list[i+1])
print(f'Resultant tuple after multiplying adjacent elements: {tuple(ans_l

```

Resultant tuple after multiplying adjacent elements: (6, 12, 20)

10) WAP to test if the given tuple is distinct or not.**Sample Input:**

```
tuple1 = (1, 2, 3, 4)
tuple2 = (1, 2, 2, 3)
```

Sample Output:

```
(1, 2, 3, 4) is distinct? -> True
(1, 2, 2, 3) is distinct? -> False
```

```
In [3]: t1 = tuple(map(int, input('Enter Tuple:').split()))

print()
for i in range(len(t1) - 1):
    if t1[i] == t1[i+1]:
        print(f'{t1} is distinct? -> False')
        break
else:
    print(f'{t1} is distinct? -> True')
```

```
(1, 2, 2, 3) is distinct? -> False
```

11) WAP to rotate the elements of a tuple to the right by k positions.**Sample Input:**

```
T = (10, 20, 30, 40, 50)
k = 2
```

Sample Output:

```
(40, 50, 10, 20, 30)
```

```
In [58]: t = tuple(map(int, input('Enter Tuple:').split()))
k = int(input('Enter K: '))

new_tuple = t[-k:] + t[0:-k]
print(new_tuple)
```

```
(30, 40, 50, 10, 20)
```

12) WAP to merge two tuples of equal length alternately.**Sample Input:**

```
T1 = (1, 3, 5)
T2 = (2, 4, 6)
```

Sample Output:

```
(1, 2, 3, 4, 5, 6)
```

```
In [6]: t1 = tuple(map(int, input('Enter Tuple:').split()))
t2 = tuple(map(int, input('Enter Tuple:').split()))
```

```
result = tuple()
print(result)

(1, 2, 3, 4, 5, 6)
```

13) WAP to create a tuple of squares of elements that are even and greater than 10.

Sample Input:

```
T = (4, 12, 7, 18, 10)
```

Sample Output:

```
(144, 324)
```

```
In [7]: t = tuple(map(int,input('Enter Tuple: ').split()))
result = (*(i**2 for i in t if i % 2 == 0 and i > 10),)
print(result)

(144, 324)
```

14) WAP to create (index, value) pairs for prime numbers.

Sample Input:

```
T = (4, 7, 9, 11, 15)
```

Sample Output:

```
((1, 7), (3, 11))
```

```
In [13]: t = tuple(map(int,input('Enter Tuple: ').split()))
index_prime_list = []
for n in t:
    for i in range(2,int(n**0.5)+1):
        if n%i == 0:
            break
    else:
        index_prime_list.append((t.index(n),n))
result_tuple = tuple(index_prime_list)
print(result_tuple)

((1, 7), (3, 11))
```

15) WAP to split a tuple into even-index and odd-index tuples.

Sample Input:

```
T = ('a', 'b', 'c', 'd', 'e')
```

Sample Input:

```
even = ('a', 'c', 'e')
odd = ('b', 'd')
```

```
In [1]: t = tuple(input('Enter Tuple: ').split())
even = tuple(t[i] for i in range(0, len(t), 2))
odd = tuple(t[i] for i in range(1, len(t), 2))

print(f'even = {even}')
print(f'odd = {odd}')
```

even = ('a', 'c', 'e')
odd = ('b', 'd')

16) WAP to compute column-wise sum from a tuple of lists.**Sample Input:**

```
T = ([1, 2, 3], [4, 5, 6], [7, 8, 9])
```

Sample Output:

```
12 15 18
```

```
In [7]: tuple_of_list = []

number_of_element = int(input('Enter Number of Element(list) in Tuple: '))

for i in range(number_of_element):
    tuple_of_list.append(list(map(int, input('Enter a list').split())))

tuple_of_list = tuple(tuple_of_list)

for j in range(len(tuple_of_list[0])):
    column_sum = 0
    for i in range(len(tuple_of_list)):
        column_sum += tuple_of_list[i][j]
    print(column_sum, end=' ')
```

```
12 15 18
```