

TUPLE ASSIGNMENT

1) Complete all the shell functions and programs given in the material. Write down your observations (1 line each).

a) Save a tuple. Print it. Delete it.

```
tup=(1,2,3)
print(tup)
del(tup)
print(tup)
```

Observation :

In this program we understood that a tuple can be saved, printed and deleted like this.

b) Save two tuples. Concatenate them. Print it without using loops.

```
tup=(1,2,3)
nxt=(10,20,30)
new=tup+nxt
print(new)
```

Observation :

In this program we understood that two or more tuples can be concatenated using + operator.

c) Suppose a tuple contains one item. Now store the same item for five times without using loop.

```
tup=(5,)
tup=tup*5
print(tup)
```

Observation :

In this program we understood that elements of tuple can be stored as many times we want by using * operator.

d) Save a tuple. Print i th to j th item without using loop.

```
tup=(1,2,3,10,20,30)
print(tup[2:4])
```

Observation :

In this program we understood that slicing can be done with tuples also.

e) Convert a tuple in a list.

```
tup=(1,2,3,10,20,30)
new=list(tup)
print(new)
```

Observation :

In this program we understood that a tuple can be converted to list by using 'list' method.

f) Find maximum and minimum item in a tuple. Find the length of a tuple.

```
tup=(1,2,3,10,20,30)
print(max(tup))
print(min(tup))
print(len(tup))
```

Observation :

In this program we understood that maximum, minimum and length of the tuple can be found by using 'max', 'min', and 'len' methods respectively.

1) Write a Python program to reverse a tuple.

```
tup=("Hello")
y=reversed(tup)
print(tuple(y))
tup=(1,2,3)
y=reversed(tup)
print(tuple(y))
```

Observation :

In this program we understood that a tuple can be reversed by 'reversed' method and can be converted back to tuple by 'tuple' method.

2) Write a Python program to count the elements in a list until an element is a tuple.

```
tup=(1,2,3,(10,20),3,4)
c=0
for i in tup:
    if isinstance(i,tuple):
        break
    c+=1
print(c)
```

Observation :

In this program we understood how 'isinstance' method is used to count number of elements before 'tuple' type element.

- 3) Write a Python program to find the index of an item of a tuple. Convert a string to a tuple. Check it for all possible parameters of index function. Check it for an item which is not present.

```
tup=("tuple index")
t=tup.index("p")
print(t)
t=tup.index("p",5)
print(t)
t=tup.index("p",1,4)
print(t)
t=tup.index("m")
print(t)
```

Observation :

In this program we understood that the index of any element in a tuple can be found by 'index' method and all its parameters.

- 4) Write a program in Python to do slicing in all possible ways with all possible parameters, providing positive and negative values for step. Also, perform slicing from start and end both.

```
tup=(2,4,6,3,5,6,8,1,3)
sl=tup[3:5]
```

```

print(sl)
sl=tup[:5]
print(sl)
sl=tup[3:]
print(sl)
sl=tup[:]
print(sl)
sl=tup[-8:-4]
print(sl)
tup=("HELLO WORLD")
sl=tup[2:9:2]
print(sl)
sl=tup[::4]
print(sl)
sl=tup[9:2:-4]
print(sl)

```

Observation :

In this program we understood the all possible combinations of slicing operation that is done on tuple.

2) Write a program in Python to do searching either linear or binary. The choice will be provided by the user.

```

def linear(t,n):
    indx=-1
    print("**** LINEAR SEARCH ****")
    for i in range(len(t)):
        if t[i]==n:
            indx=i
            print("Element found at position ",indx+1)
            break
    if indx== -1:
        print("Not found")

def binary(t,n):
    indx=-1

```

```

print("*** BINARY SEARCH ***")
t=sorted(t)
print("List after sorting for binary search: ",t)
ll=0
ul=len(t)-1
while ll<=ul:
    mid=(ul+ll)//2
    if t[mid]==number:
        indx=mid
        print("Element found at position ",indx+1)
        break
    else:
        if t[mid]<number:
            ll=mid;
        else:
            ul=mid;
if indx==-1:
    print("Not found")

```

```

input_str = input("enter elements with space:")
lst = input_str.split()
print(lst)
t=tuple(lst)
print("The list: ",t)
n=int(input("Enter the number to search: "))
ch=int(input("Enter 1 for linear search and 2 for Binary search : "))
if ch==1:
    linear(t,n)
elif ch==2:
    binary(t,n)
else:
    print("Wrong input")

```

Output :

```

enter elements with space:1 3 2
['1', '3', '2']
The list: ('1', '3', '2')

```

Enter the number to search: 3

Enter 1 for linear search and 2 for Binary search : 1

*** LINEAR SEARCH ***

Element found at position 2