



---

# TT DS PYTHON MODULE-19



**State** Finished

**Completed on** Wednesday, 19 March 2025, 9:53 AM

**Time taken** 38 mins 9 secs

**Grade** 80.00 out of 100.00

## Question 1

Correct

Mark 20.00 out of 20.00

Flag question

Write a python program to implement merge sort using iterative approach on the given list of values.

**For example:**

Test	Input	Result
Merge_Sort(S)	6 4 2 3 1 6 5	The Original array is: [4, 2, 3, 1, 6, 5] Array after sorting is: [1, 2, 3, 4, 5, 6]
Merge_Sort(S)	5 2 6 4 3 1	The Original array is: [2, 6, 4, 3, 1] Array after sorting is: [1, 2, 3, 4, 6]

**Answer:** (penalty regime: 0 %)

```

1 def Merge_Sort(S):
2     if len(S)<2:
3         return S
4     result=[]
5     mid=int(len(S)//2
6     i=0
7     j=0
8
9     y=Merge_Sort(S[:mid])
10    z=Merge_Sort(S[mid:])
11
12    while i<len(y) and j<len(z):
13        if y[i]>z[j]:
14            result.append(z[j])
15            j+=1
16        else:
17            result.append(y[i])
18            i+=1
19
20    result+=y[i:]
21    result+=z[j:]
22    return result

```

Test	Input	Expected	Got
Merge_Sort(S)	6 4 2 3 1 6 5	The Original array is: [4, 2, 3, 1, 6, 5] Array after sorting is: [1, 2, 3, 4, 5, 6]	The Original array is: [4, 2, 3, 1, 6, 5] Array after sorting is: [1, 2, 3, 4, 5, 6]
Merge_Sort(S)	5 2 6 4 3 1	The Original array is: [2, 6, 4, 3, 1] Array after sorting is: [1, 2, 3, 4, 6]	The Original array is: [2, 6, 4, 3, 1] Array after sorting is: [1, 2, 3, 4, 6]
Merge_Sort(S)	4 3 5 6 1	The Original array is: [3, 5, 6, 1] Array after sorting is: [1, 3, 5, 6]	The Original array is: [3, 5, 6, 1] Array after sorting is: [1, 3, 5, 6]

Passed all tests!

**Correct**

Marks for this submission: 20.00/20.00.

## Question 2

Correct

Mark 20.00 out of 20.00

**Write a Python Program to print factorial of a number recursively.**



For example:

Input	Result
5	Factorial of number 5 = 120
6	Factorial of number 6 = 720

Answer: (penalty regime: 0 %)

```

1 n=int(input())
2 mul=1
3 for i in range(1,n+1):
4     mul*=i
5
6 print(f"Factorial of number {n} =",mul)
7

```

	Input	Expected	Got	
	5	Factorial of number 5 = 120	Factorial of number 5 = 120	
	6	Factorial of number 6 = 720	Factorial of number 6 = 720	
	7	Factorial of number 7 = 5040	Factorial of number 7 = 5040	
	8	Factorial of number 8 = 40320	Factorial of number 8 = 40320	

Passed all tests!

Correct

Marks for this submission: 20.00/20.00.

Question 3

Correct

Mark 20.00 out of 20.00

Flag question

Write a python program to implement linear search on the given tuple of string values.

note: As the tuple is immutable convert the list to tuple to perform search

For example:

Input	Result
5 ram john akbar seetha oviya john	Tuple: john found
4 rohini fathima jenifer nizam rakesh	Tuple: rakesh not found

Answer: (penalty regime: 0 %)

```

1 def search(Tuple,x):
2     for i in range(len(Tuple)):
3         if Tuple[i]==x:
4             return True
5     return False
6
7
8
9

```



```

12 for i in range(n):
13     arr.append(input())
14 x=input()
15 Tuple=tuple(arr)
16 if search(Tuple,x):
17     print(f"Tuple: {x} found")
18 else:
19     print(f"Tuple: {x} not found")

```

	Input	Expected	Got	
	5 ram john akbar seetha oviya john	Tuple: john found	Tuple: john found	
	4 rohini fathima jenifer nizam rakesh	Tuple: rakesh not found	Tuple: rakesh not found	
	6 rose jasmine tulips marigold hibiscus lotus lilly	Tuple: lilly not found	Tuple: lilly not found	

Passed all tests!

Marks for this submission: 20.00/20.00.

#### Question 4

Incorrect

Mark 0.00 out of 20.00

Flag question

Write a python program to implement quick sort on the given float array values.

For example:

Input	Result
5 6.9 8.3 2.1 1.5 6.4	left: [] right: [] left: [] right: [] left: [1.5] right: [6.4] left: [] right: [] left: [1.5, 2.1, 6.4] right: [8.3] [1.5, 2.1, 6.4, 6.9, 8.3]
6 3.1 2.4 5.6 4.3 6.2 7.8	left: [] right: [] left: [] right: [] left: [] right: [] left: [] right: [7.8] left: [4.3] right: [6.2, 7.8] left: [2.4] right: [4.3, 5.6, 6.2, 7.8] [2.4, 3.1, 4.3, 5.6, 6.2, 7.8]

Answer: (penalty regime: 0 %)

```

1 def quicksort(arr,l,r):
2     if r-l>1:
3         p=partition(arr,l,r)
4         quicksort(arr,l,p)
5         quicksort(arr,p+1,r)
6
7 def partition(arr,l,r):
8     pivot=arr[l]
9     i=l+1
10    j=r-1

```



```

14         i+=1
15         while i<=j and arr[j]>=pivot:
16             j-=1
17
18     if i<=j :
19         arr[i],arr[j]=arr[j],arr[i]
20     else:
21         arr[j],arr[1]=arr[1],arr[j]
22     return j

```

Input	Expected	Got
5 6.9 8.3 2.1 1.5 6.4	left: [] right: [] left: [] right: [] left: [1.5] right: [6.4] left: [] right: [] left: [1.5, 2.1, 6.4] right: [8.3] [1.5, 2.1, 6.4, 6.9, 8.3]	***Time limit exceeded***

Testing was aborted due to error.  
Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect

Marks for this submission: 0.00/20.00.

#### Question 5

Correct

Mark 20.00 out of 20.00

Flag question

Write a python program for a search function with parameter list name and the value to be searched on the given list of int values.

For example:

Test	Input	Result
search(List, n)	5 3 4 5 6 7 4	Found
search(List, n)	6 20 34 56 87 96 51 87	Found

Answer: (penalty regime: 0 %)

```

1 def search(List,n):
2     for i in range(len(List)):
3         if List[i]==n:
4             return True
5     return False
6
7
8
9
10 List=[]
11 x=int(input())
12 for i in range(x):
13     List.append(int(input()))
14 n=int(input())
15 if search(List,n):
16     print("Found")
17 else:
18     print("Not Found")

```



	search(List, n)	5 3 4 5 6 7 4	Found	Found	
	search(List, n)	6 20 34 56 87 96 51 87	Found	Found	
	search(List, n)	4 30 10 20 50 60	Not Found	Not Found	

Passed all tests!



Marks for this submission: 20.00/20.00.