Started on	Thursday, 4 April 2024, 3:06 PM
State	Finished
Completed on	Thursday, 4 April 2024, 3:43 PM
Time taken	36 mins 36 secs
Grade	**100.00** out of 100.00

Question **1**

Correct

Mark 20.00 out of 20.00

Write a python program to implement the quick sort using recursion on the given list of float values.

For example:

Input	Result
5 6.3 1.2 4.6 5.8 9.7	pivot: 9.7 pivot: 5.8 pivot: 4.6 [1.2, 4.6, 5.8, 6.3, 9.7]
6 2.3 7.8 9.5 4.2 3.6 5.4	pivot: 5.4 pivot: 3.6 pivot: 7.8 [2.3, 3.6, 4.2, 5.4, 7.8, 9.5]

Answer: (penalty regime: 0 %)

```
def part(arr,l,r):
    pi=arr[r]
    i=l-1

    for j in range(l,r):
        if arr[j]<=pi:
            i=i+1
            arr[i],arr[j]=arr[j],arr[i]
    arr[i+1],arr[r]=arr[r],arr[i+1]
    return i+1
def quickSort(arr,l,r):
    if l<r:
        p=part(arr,l,r)
        print("pivot: ",arr[p])
        quickSort(arr,l,p-1)
        quickSort(arr,p+1,r)
    return arr
arr=list()
n=int(input())
for i in range(0,n):
    ele=float(input())
    arr.append(ele)
```

	Input	Expected	Got	
✔	5 6.3 1.2 4.6 5.8 9.7	pivot: 9.7 pivot: 5.8 pivot: 4.6 [1.2, 4.6, 5.8, 6.3, 9.7]	pivot: 9.7 pivot: 5.8 pivot: 4.6 [1.2, 4.6, 5.8, 6.3, 9.7]	✔
✔	6 2.3 7.8 9.5 4.2 3.6 5.4	pivot: 5.4 pivot: 3.6 pivot: 7.8 [2.3, 3.6, 4.2, 5.4, 7.8, 9.5]	pivot: 5.4 pivot: 3.6 pivot: 7.8 [2.3, 3.6, 4.2, 5.4, 7.8, 9.5]	✔

	Input	Expected	Got	
✔	4 3.2 6.4 8.7 1.5	pivot: 1.5 pivot: 3.2 pivot: 6.4 [1.5, 3.2, 6.4, 8.7]	pivot: 1.5 pivot: 3.2 pivot: 6.4 [1.5, 3.2, 6.4, 8.7]	✔

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 implement linear search on the given tuple of float values.

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

For example:

Input	Result
5 3.2 1.5 6.4 7.8 9.5 6.4	Tuple: 6.4 found
6 3.2 1.2 3.4 5.3 6.2 6.8 6.2	Tuple: 6.2 found

Answer: (penalty regime: 0 %)

```
def LS(list,n,key):
    for i in range(0,n):
        if(list[i]==key):
            return i
    return -1
list=[]
n=int(input())
for i in range(0,n):
    temp=input()
    list.append(temp)
key=input()
res=LS(list,n,key)
if(res==-1):
    print("Tuple: %s not found"%key)
else:
    print("Tuple: %s found"%key)
```

	Input	Expected	Got	
✔	5 3.2 1.5 6.4 7.8 9.5 6.4	Tuple: 6.4 found	Tuple: 6.4 found	✔

	Input	Expected	Got	
✔	6 3.2 1.2 3.4 5.3 6.2 6.8 6.2	Tuple: 6.2 found	Tuple: 6.2 found	✔
✔	4 2.1 3.2 6.5 4.5 3.5	Tuple: 3.5 not found	Tuple: 3.5 not found	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Correct

Mark 20.00 out of 20.00

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 %)

```
def Merge_Sort(S):
    size=len(S)
    if(size>1):
        mid=size//2
        leftarr=S[:mid]
        rightarr=S[mid:]
        Merge_Sort(leftarr)
        Merge_Sort(rightarr)
        i=j=k=0
        while(i<len(leftarr) and j<len(rightarr)):
            if(leftarr[i]<rightarr[j]):
                S[k]=leftarr[i]
                k+=1
                i+=1
            else:
                S[k]=rightarr[j]
                k+=1
                j+=1
        while(i<len(leftarr)):
            S[k]=leftarr[i]
            k+=1
            i+=1
```

	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]	✔

	Test	Input	Expected	Got	
✔	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 **4**

Correct

Mark 20.00 out of 20.00

Write a python program to implement binary search on the given list of string values using iterative method

For example:

Test	Input	Result
`binarySearchAppr(arr, 0, len(arr)-1, x)`	5 one two three four five two	`Element is present at index 4`
`binarySearchAppr(arr, 0, len(arr)-1, x)`	6 one three five seven nine eleven thirteen	`Element is not present in array`

Answer: (penalty regime: 0 %)

```
def binarySearchAppr(arr, start, end, x):
    if end >= start:
        mid = (start + end) // 2
        if arr[mid] == x:
            return mid
        elif arr[mid] > x:
            return binarySearchAppr(arr, start, mid-1, x)
        else:
            return binarySearchAppr(arr, mid+1, end, x)
    else:
        return -1

arr = []
n = int(input(""))
for i in range(n):
    arr.append(input())

arr = sorted(arr)
x = input()

result = binarySearchAppr(arr, 0, len(arr)-1, x)

```

	Test	Input	Expected	Got	
✔	`binarySearchAppr(arr, 0, len(arr)-1, x)`	5 one two three four five two	`Element is present at index 4`	`Element is present at index 4`	✔
✔	`binarySearchAppr(arr, 0, len(arr)-1, x)`	6 one three five seven nine eleven thirteen	`Element is not present in array`	`Element is not present in array`	✔

	Test	Input	Expected	Got	

✔	`binarySearchAppr(arr, 0, len(arr)-1, x)`	`4`
		`two`
		`four`
		`six`
		`eight`
		`six`
			`Element is present at index 2`	`Element is present at index 2`	✔

Passed all tests! ✔

Correct

Marks for this submission: 20.00/20.00.

Question **5**

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 %)

```
n=int(input())
for i in range(1,n):
    n=n*i
print("Factorial of number",i+1,"=",n)

```

	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.