Started on	Wednesday, 19 March 2025, 9:15 AM
State	Finished
Completed on	Wednesday, 19 March 2025, 9:58 AM
Time taken	42 mins 35 secs
Grade	**80.00** out of 100.00

Question **1**

Correct

Mark 20.00 out of 20.00

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 %)

```
def qsort(L):
    if L==[]:
        return []
    pivot=L[0:1]
    left=qsort([x for x in L[1:] if x<L[0]])
    right=qsort([x for x in L[1:] if x>L[0]])
    print('left: ',left)
    print('right: ',right)
    return left+pivot+right
L=[float(input()) for i in range(int(input()))]
L=qsort(L)
print(L)
```

	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]	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]	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]	✓
✓	8 1.2 1.3 4.2 5.3 6.4 7.3 6.8 9.2	left: [] right: [] left: [] right: [] left: [6.8] right: [9.2] left: [] right: [6.8, 7.3, 9.2] left: [] right: [6.4, 6.8, 7.3, 9.2] left: [] right: [5.3, 6.4, 6.8, 7.3, 9.2] left: [] right: [4.2, 5.3, 6.4, 6.8, 7.3, 9.2] left: [] right: [1.3, 4.2, 5.3, 6.4, 6.8, 7.3, 9.2] [1.2, 1.3, 4.2, 5.3, 6.4, 6.8, 7.3, 9.2]	left: [] right: [] left: [] right: [] left: [6.8] right: [9.2] left: [] right: [6.8, 7.3, 9.2] left: [] right: [6.4, 6.8, 7.3, 9.2] left: [] right: [5.3, 6.4, 6.8, 7.3, 9.2] left: [] right: [4.2, 5.3, 6.4, 6.8, 7.3, 9.2] left: [] right: [1.3, 4.2, 5.3, 6.4, 6.8, 7.3, 9.2] [1.2, 1.3, 4.2, 5.3, 6.4, 6.8, 7.3, 9.2]	✓

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 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 %)

```python
def search(List,n):
    for i in List:
        if i==n:
            print('Found')
            return
    print("Not Found")
List=[int(input()) for i in range(int(input()))]
n=int(input())


```

	Test	Input	Expected	Got	
✓	`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! ✓

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Incorrect

Mark 0.00 out of 20.00

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

For example:

Test	Input	Result

`Merge_Sort(S)`	5
	10.2
	21.3
	3.5
	7.8
	9.8
		`The Original array is:  [10.2, 21.3, 3.5, 7.8, 9.8]`
		`Array after sorting is:  [3.5, 7.8, 9.8, 10.2, 21.3]`

`Merge_Sort(S)`	6
	20.3
	41.2
	5.3
	6.2
	8.1
	65.2
		`The Original array is:  [20.3, 41.2, 5.3, 6.2, 8.1, 65.2]`
		`Array after sorting is:  [5.3, 6.2, 8.1, 20.3, 41.2, 65.2]`

Answer: (penalty regime: 0 %)

```python
def Merge_Sort(S):
    if len(S)>1:
        mid=len(S)//2
        la=S[:mid]
        ra=S[mid:]
        Merge_Sort(la)
        Merge_Sort(ra)
        ls=len(la)
        rs=len(ra)
        p,q,r=0,0,0
        while p<ls and q<ls:
            if la[p]<ra[q]:
                S[r]=la[p]
                p+=1
            else:
                S[r]=ra[q]
                q+=1
            r+=1
        while p<ls:
            S[r]=la[p]
            p+=1
            r+=1
        while q<rs:
            S[r]=ra[q]
            q+=1
```

	Test	Input	Expected	Got	

✗	`Merge_Sort(S)`	5
		10.2
		21.3
		3.5
		7.8
		9.8
			`The Original array is:  [10.2, 21.3, 3.5, 7.8, 9.8]`
			`Array after sorting is:  [3.5, 7.8, 9.8, 10.2, 21.3]`
				`The Original array is: [10.2, 21.3, 3.5, 7.8, 9.8]`
				`Array after sorting is: [3.5, 7.8, 10.2, 21.3, 9.8]`
					✗

Some hidden test cases failed, too.

Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect

Marks for this submission: 0.00/20.00.

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 %)

```python
def binarySearchAppr(arr,l,r,x):
    while l<=r:
        mid=(l+r)//2
        if arr[mid]==x:
            return mid
        elif x>arr[mid]:
            l=mid+1
        else:
            r=mid-1
    return -1
arr=[input() for i in range(int(input()))]
x=input()
arr.sort()
r=binarySearchAppr(arr,0,len(arr)-1,x)
if r==-1:
    print("Element is not present in array")
else:
    print("Element is present at index",r)
```

	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	✓

	Test	Input	Expected	Got	
✓	`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`	✓
✓	`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 Using a recursive function to calculate the sum of a sequence
For example:

Input	Result
20	210
36	666
45	1035

Answer: (penalty regime: 0 %)

```
a=int(input())
sum=0
for i in range(1,a+1):
    sum+=i
print(sum)
```

	Input	Expected	Got	
✓	20	210	210	✓
✓	36	666	666	✓
✓	45	1035	1035	✓
✓	58	1711	1711	✓
✓	65	2145	2145	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.