Started on	Friday, 21 March 2025, 3:43 PM
State	Finished
Completed on	Friday, 21 March 2025, 3:57 PM
Time taken	13 mins 51 secs
Grade	**80.00** out of 100.00

Question **1**

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 %)

```python
def sumof(n,a,result):
    if (n<a):
        print (result)
    else:
        result+=a
        sumof(n,a+1,result)
n=int(input())
sumof(n,1,0)
```

	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.

Question **2**

Correct

Mark 20.00 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):
    n = len(S)
    current_size = 1

    while current_size < n:
        left = 0
        while left < n - 1:
            mid = min(left + current_size - 1, n - 1)
            right = min(left + 2 * current_size - 1, n - 1)

            merge(S, left, mid, right)
            left += 2 * current_size

        current_size *= 2


def merge(S, left, mid, right):
    n1 = mid - left + 1
    n2 = right - mid

    L = [S[left + i] for i in range(n1)]
    R = [S[mid + 1 + i] for i in range(n2)]
```

	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, 9.8, 10.2, 21.3]	✓

	Test	Input	Expected	Got	
✓	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]	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]	✓
✓	Merge_Sort(S)	4 2.3 6.1 4.5 96.5	The Original array is: [2.3, 6.1, 4.5, 96.5] Array after sorting is: [2.3, 4.5, 6.1, 96.5]	The Original array is: [2.3, 6.1, 4.5, 96.5] Array after sorting is: [2.3, 4.5, 6.1, 96.5]	✓

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 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 %)

```python
def partition(arr,low,high):
    pivot=arr[high]
    i=low-1
    for j in range(low,high):
        if arr[j]<=pivot:
            i+=1
            arr[i],arr[j]=arr[j],arr[i]
    arr[i+1],arr[high]=arr[high],arr[i+1]
    return i+1
def quick_sort(arr,low,high):
    if low<high:
        pivot_index=partition(arr,low,high)
        print("pivot: ",arr[pivot_index])
        quick_sort(arr,low,pivot_index-1)
        quick_sort(arr,pivot_index+1,high)
n=int(input())
arr=[]
for i in range(n):
    element=float(input())
    arr.append(element)
quick_sort(arr,0,len(arr)-1)
print(arr)
```

	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]	✓

	Input	Expected	Got	
✓	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]	✓
✓	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 **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(list1,low,high,x):
    mid = 0
    while(low<=high):
        mid = (low+high)//2
        if list1[mid] < x:
            low = mid  + 1
        elif list1[mid] > x:
            high = mid - 1
        else:
            return mid
    return -1
arr = []
num = int(input())
for i in range(0,num):
    arr.append(input())
arr = sorted(arr)
x = input()
value = binarySearchAppr(arr, 0, len(arr)-1, x)
if value==-1:
    print("Element is not present in array")
else:
    print("Element is present at index",str(value))
```

	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**

Not answered

Mark 0.00 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 %)