Started on	Thursday, 17 April 2025, 11:11 AM
State	Finished
Completed on	Thursday, 17 April 2025, 11:29 AM
Time taken	18 mins 27 secs
Grade	**80.00** out of 100.00

Question **1**

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 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
global key
def search(List,n):
    for i in range(n):
        if(List[i]==key):
            return i
    else:
        return -1
List=[]
n=int(input())
for i in range(n):
    List.append(int(input()))
key=int(input())
res=search(List,n)
if(res!=-1):
    print("Found")
else:
    print("Not Found")
```

	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 **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 float values.

For example:

Test	Input	Result
search(List, n)	5 3.2 6.1 4.5 6.2 8.5 3.2	3.2 Found
search(List, n)	4 3.2 1.5 6.4 7.8 6.1	6.1 Not Found

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
global key
def search(List,n):
    for i in range(n):
        if(List[i]==key):
            return i
    else:
        return -1
List=[]
n=int(int(input()))
for i in range(n):
    List.append(float(input()))
key=float(input())
res=search(List,n)
if(res!=-1):
    print(f"{key} Found")
else:
    print(f"{key} Not Found")
```

	Test	Input	Expected	Got	
✓	search(List, n)	5 3.2 6.1 4.5 6.2 8.5 3.2	3.2 Found	3.2 Found	✓

	Test	Input	Expected	Got	
✓	`search(List, n)`	4 3.2 1.5 6.4 7.8 6.1	6.1 Not Found	6.1 Not Found	✓
✓	`search(List, n)`	7 2.1 3.2 6.5 4.1 5.2 7.1 8.2 9.3	9.3 Not Found	9.3 Not Found	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Not answered

Mark 0.00 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 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

Question **4**

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 %)

Ace editor not ready. Perhaps reload page?

Falling back to raw text area.

```
def Merge_Sort(S):
    size=len(S)
    if(size>1):
        mid=size//2
        l=S[:mid]
        r=S[mid:]
        Merge_Sort(l)
        Merge_Sort(r)
        ls=len(l)
        rs=len(r)
        i=j=k=0
        while(i<ls and j<rs):
            if(l[i]<r[j]):
                S[k]=l[i]
                i+=1
            else:
                S[k]=r[j]
                j+=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, 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 **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 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
def fact(n):
    if(n==0):
        return 1
    else:
        return n*fact(n-1)
n=int(input())
print(f"Factorial of number {n} = {fact(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.