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 lin_search(arr,key):
    flag=0
    for i in arr:
        if key==i:
            flag=1
            print(f"Tuple: {key} found")
    if flag==0:
        print(f"Tuple: {key} not found")


n=int(input())
arr=[]
for i in range(n):
    num=float(input())
    arr.append(num)
key=float(input())
lin_search(arr,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	
	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 **2**

Correct

Mark 20.00 out of 20.00

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

Flag question

For example:

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

Answer: (penalty regime: 0 %)

```
def factorial(n):
    if n==1:
        return 1
    else:
        return n*factorial(n-1)

n=int(input())
print(f"Factorial of number {n} = {factorial(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.

Question **3**

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 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 %)

```
def Merge_Sort(S):
    n=len(S)
    size=1
    while size<n:
        low=0
        while low<n:
            mid=min(low+size-1,n-1)
            high=min(low+(2*size)-1,n-1)
            merge(S,low,mid,high)
```

```
            merge(S,low,mid,high)
            low+=2*size
        size*=2

def merge(arr,low,mid,high):
    left=low
    right=mid+1
    temp=[]
    while left<=mid and right<=high:
        if arr[left]<arr[right]:
            temp.append(arr[left])
            left+=1
        else:
            temp.append(arr[right])
```

	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, Array after sorting is: [3.5, 7.8,
	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, Array after sorting is: [5.3, 6.2,
	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 Array after sorting is: [2.3, 4.5,

Passed all tests!

Correct

Marks for this submission: 20.00/20.00.

Question **4**

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 %)

```
def lin_search(arr,key):
    flag=0
    for i in arr:
        if i==key:
            print(f"Tuple: {key} found")
            flag=1
    if flag==0:
        print(f"Tuple: {key} not found")

n=int(input())
arr=[]
for i in range(n):
    name=input()
    arr.append(name)
key=input()
lin_search(arr,key)
```

	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!

Correct

Marks for this submission: 20.00/20.00.

Question **5**

Incorrect

Mark 0.00 out of 20.00

Flag question

Write a python program to implement quick sort on the given float values and print the sorted list and pivot value of each iteration.

For example:

Input	Result
5 2.3 3.2 1.6 4.2 3.9	Input List [2.3, 3.2, 1.6, 4.2, 3.9] pivot: 2.3 pivot: 3.2 pivot: 4.2 Sorted List [1.6, 2.3, 3.2, 3.9, 4.2]
4 5 2 49 3	Input List [5.0, 2.0, 49.0, 3.0] pivot: 5.0 pivot: 3.0 Sorted List [2.0, 3.0, 5.0, 49.0]

Answer: (penalty regime: 0 %)

```
def quick_sort(arr):
    mid=len(arr)//2
    pivot=arr[mid]
    print("pivot: ",pivot)
    for i in arr:
        if arr[i]>pivot:
            arr[mid]=arr[i]

n=int(input())
arr=[]
for i in range(n):
    num=float(input())
    arr.append(num)
print("Input List")
print(arr)
quick_sort(arr)
print("Sorted List")
print(arr)
```

	Input	Expected	Got	
	5 2.3 3.2 1.6 4.2 3.9	Input List [2.3, 3.2, 1.6, 4.2, 3.9] pivot: 2.3 pivot: 3.2 pivot: 4.2 Sorted List [1.6, 2.3, 3.2, 3.9, 4.2]	Input List [2.3, 3.2, 1.6, 4.2, 3.9] pivot: 1.6 ***Run error*** Traceback (most recent call last): File "__tester__.python3", line 16, in <module> quick_sort(arr) File "__tester__.python3", line 6, in quick_sort if arr[i]>pivot: TypeError: list indices must be integers or slices, not float	

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.