


Branch: master

Find file

Copy path

Apssdc-Python / 26-June-2019.ipynb

 **SireeshaReyyi** 28 June 2019  
dfe809f 4 days ago

1 contributor

<> 

RawBlameHistory

475 lines (474 sloc) 12.3 KB

```

In [41]: # Largest number in a List
# Second Largest number in a List
# Kth Largest number in a List

# Element with highest frequency
# Second highest frequency
# Kth highest frequency

# Function to find the second largest number in a List
def secondLargest(li):
    # Convert the List into a unique List
    unique = []
    for n in li:
        if n not in unique:
            unique.append(n)
    # unique.sort()
    # unique = unique[-1::-1]
    if len(unique) == 1:
        print("There is only one element in the given list")
    else:
        unique = sorted(unique, reverse=True)
        return unique[1]

# Function to find the fifth smallest number in a List
def fifthLeast(li):
    unique = []
    for i in li:
        unique.append(i)
    unique.sort()
    if len(unique) < 5:
        return -1
    else:
        return li[4]

# Function to find the Kth Largest number in a List
def kLargest(li,k):
    unique = []
    for i in li:
        if i not in unique:
            unique.append(i)
    unique = sorted(unique,reverse=True)
    if len(unique)<k:
        return -1
    else:
        return unique[k-1]

# Function to find the Kth smallest number in a List
def kSmallest(li,k):
    unique = []
    for i in li:
        if i not in unique:
            unique.append(i)
    unique.sort()
    if k== 0:
        return -1
    if len(unique)<k:
        return -1
    else:
        return unique[k-1]

li=[1,6,4,3,5,9,6,1,1,5,8,9,2,3,4,6]
k=int(input("enter k : "))
secondLargest(li)
fifthLeast(li)
kLargest(li,k)
kSmallest(li,k)

```

enter k : 2

Out[41]: 2

```

In [72]: # Function to identify the element with highest frequency
# highestFrequency([1,2,3,9,8,7,3,4,2,1]) -> if two elements have same highest frequency , return smallest

def highestFrequencyElement(li,k):
    unique = []

```

```

    for i in li:
        if i not in unique:
            unique.append(i)
    freq = []
    for i in unique:
        freq.append(li.count(i))
    freq = sorted(freq,reverse=True)
    unique2 = []
    for i in freq:
        if i not in unique2:
            unique2.append(i)
    elements = []
    for i in unique:
        if unique2[0]==li.count(i):
            elements.append(i)
    elements = sorted(elements,reverse=True)
    if k>len(elements):
        return -1
    return elements[k-1]

def highestFrequencyElement2(li,k):
    unique = {}
    for n in li:
        if n not in unique:
            unique[n] = 1
        else:
            unique[n] += 1
    # Getting all frequencies into a list
    freq = unique.values()
    maxfreq = max(freq)
    maxfreqKeys = []
    # Identify the keys maximum frequency
    for item in unique.items():
        if item[1] == maxfreq:
            maxfreqKeys.append(item[0])
    # Select the minimum from the keys with maximum frequency
    maxfreqKeys = sorted(maxfreqKeys,reverse=True)
    if k>len(maxfreqKeys):
        return -1
    return maxfreqKeys[k-1]

li = [1,2,3,9,8,7,3,4,2,1]
k=int(input())
highestFrequencyElement(li,k)
highestFrequencyElement2(li,k)

```

```

4
dict_items([(1, 2), (2, 2), (3, 2), (9, 1), (8, 1), (7, 1), (4, 1)])

```

Out[72]: -1

```

In [93]: # Function to identify second largest frequency element
# If there are many such elements, return the smallest
# li = [1,2,3,2,1,4,4,9]
def secondLargestFrequency(li):
    unique = {}
    for i in li:
        if i not in unique:
            unique[i] = 1
        else:
            unique[i] += 1
    freq = unique.values()
    uniquefreq = []
    for i in freq:
        if i not in uniquefreq:
            uniquefreq.append(i)
    uniquefreq = sorted(uniquefreq,reverse=True)
    elements = []
    for item in unique.items():
        if item[1] == uniquefreq[1]:
            elements.append(item[0])
    elements = sorted(elements,reverse=False)
    return elements[0]

li = [1,2,3,2,1,4,4,9]
secondLargestFrequency(li)

```

Out[93]: 5

```
In [99]: # Function to identify Kth largest frequency element
# If there are many such elements, return the smallest
# li = [9,8,7,6,5,2,3,4,9,6,7,7,7,6,7,6], k=4 -> 2
def kLargestFrequency(li,k):
    unique = {}
    for i in li:
        if i not in unique:
            unique[i] = 1
        else:
            unique[i] += 1
    freq = unique.values()
    uniquefreq = []
    for i in freq:
        if i not in uniquefreq:
            uniquefreq.append(i)
    uniquefreq = sorted(uniquefreq,reverse=True)
    elements = []
    if len(uniquefreq)>=k:
        for item in unique.items():
            if item[1] == uniquefreq[k-1]:
                elements.append(item[0])
        elements = sorted(elements,reverse=False)
        return elements[0]
    else:
        return -1

li = [9,8,7,6,5,2,3,4,9,6,7,7,7,6,7,6]
k=int(input())
kLargestFrequency(li,k)
```

5

Out[99]: -1

```
In [101]: # Function to identify Kth lowest frequency element
# If there are many such elements, return the smallest
# li = [9,8,7,6,5,2,3,4,9,6,7,7,7,6,7,6], k=4 -> 2
def kLowestFrequency(li,k):
    unique = {}
    for i in li:
        if i not in unique:
            unique[i] = 1
        else:
            unique[i] += 1
    freq = unique.values()
    uniquefreq = []
    for i in freq:
        if i not in uniquefreq:
            uniquefreq.append(i)
    uniquefreq = sorted(uniquefreq,reverse=False)
    print(uniquefreq)
    elements = []
    if len(uniquefreq)>=k:
        for item in unique.items():
            if item[1] == uniquefreq[k-1]:
                elements.append(item[0])
        elements = sorted(elements,reverse=False)
        return elements[0]
    else:
        return -1

li = [9,8,7,6,5,2,3,4,9,6,7,7,7,6,7,6]
k=int(input())
kLowestFrequency(li,k)
```

4

[1, 2, 4, 5]

Out[101]: 7

```
In [4]: def kLargestFrequencyString(s,k):
        unique = {}
        for i in s:
            if i not in unique:
                unique[i] = 1
            else:
                unique[i] += 1
        freq = unique.values()
```

```

uniquefreq = []
for i in freq:
    if i not in uniquefreq:
        uniquefreq.append(i)
uniquefreq = sorted(uniquefreq,reverse=True)
elements = []
if k<len(uniquefreq):
    for item in unique.items():
        if item[1] == uniquefreq[k-1]:
            elements.append(item[0])
    elements = sorted(elements,reverse=False)
    return elements[0]
else:
    return -1
filepath = 'DataFiles/Kth_Largest-Frequency.txt'
with open(filepath,'r') as f:
    t=int(f.readline())
    for i in range(t):
        s=f.readline()
        k=int(f.readline())
        print(kLargestFrequencyString(s,k))

```

```

s
g
h
e
w
r
n
k
-1
-1

```

```

In [2]: def kLargestFrequencyString(s,k):
        unique = {}
        for i in s:
            if i not in unique:
                unique[i] = 1
            else:
                unique[i] += 1
        freq = unique.values()
        uniquefreq = []
        for i in freq:
            if i not in uniquefreq:
                uniquefreq.append(i)
        uniquefreq = sorted(uniquefreq,reverse=True)
        elements = []
        if k<=len(uniquefreq):
            for item in unique.items():
                if item[1] == uniquefreq[k-1]:
                    elements.append(item[0])
            elements = sorted(elements,reverse=False)
            return elements[0]
        else:
            return -1

        s='abcdefbcdeab'
        k=3
        kLargestFrequencyString(s,k)

```

Out[2]: 'f'

In [ ]: