In [5]:

```
###### {a:4,g:9,i:6,p:213,c=6}
# [4,6,6,9,213]
# [a,c,g,i,p]
# k=3
# li=[]
# for item in d.items():
# if item[1]==6:
#       li.append(item[0])
# li=[i,c]

def kLargestFrequency(s,k):
    #Construct the frequency dic
    unique=[]
    freq={}
    for i in s:
        if i not in unique:
            freq[i]=s.count(i)
    values=sorted(freq.values(),reverse=True)
    uniquevalues=list(set(values))
    uniquevalues=sorted(uniquevalues,reverse=True)
    if len(uniquevalues)>=k:
        kvalue=uniquevalues[k-1]
    else:
        return -1
    for item in freq.items():
        if item[1]==kvalue:
            li.append
    return min(li)
print(kLargestFrequency([1,2,3,4,2,1,3],4))
```

```
-1
```

In [7]:

```python
# Function to find second largest number

def secondLargest(li):
    #Convert the list into a unique list
    unique=[]
    for n in li:
        if n not in unique:
            unique.append(n)
    unique=sorted(unique,reverse=True)
    return unique[1]
secondLargest([1,2,3,4,5,8,7,1,2,3])

def fifthLeast(li):
    unique=[]
    for n in li:
        if n not in unique:
            unique.append(n)
    unique=sorted(unique,reverse=True)
    return unique[4]
fifthLeast([1,2,3,4,5,8,7,1,2,3])

def secondLargest(li):
    unique=[]
    for n in li:
        if n not in unique:
            unique.append(n)
    unique=sorted(unique,reverse=True)
    if len(unique)>=1:
        return unique[1]
    return -1
secondLargest([1,2])

def kLargest(li):
    unique=[]
    k=int(input())
    for n in li:
        if n not in unique:
            unique.append(n)
    unique=sorted(unique,reverse=True)
    if len(unique)>=k:
        return unique[k-1]
    return -1
kLargest([1,2,3,4,12,4,5,7,5])
```

```
3
```

Out[7]:

```
5
```

In [8]:

```
def kSmallest(li):
    k=int(input())
    unique=[]
    for n in li:
        if n not in unique:
            unique.append(n)
    unique=sorted(unique)
    if len(unique)>=k:
        return unique[k-1]
    return -1
kSmallest([3,3,9])
```

```
2
```

Out[8]:

```
9
```

In [9]:

```
# Function to identify the element with highest frequency in a list
# If many elements have the highest frequency returns the smallest
# Highest frequency of following elements([1,2,3,9,8,7,3,4,2,1])->1

def highestFrequencyElement(li):
    unique={}
    for i in li:
        if i in unique:
            unique[i]+=1
        else:
            unique[i]=1
    #unique={1:2,2:2,3:2,9:1,8:1,7:,4:1}

    #Getting all frequencies into list
    freq=unique.values()
    maxfreq=max(freq)

    #Extract all the keys with maxfreq in a list
    maxfreqkeys=[]

    #Identify the smallest number
    for item in unique.items():
        if item[1]==maxfreq:
            maxfreqkeys.append(item[0])

    # Select the minimum from the keys with values

    return min(maxfreqkeys) #minfreqE[k-1]

print(highestFrequencyElement([1,2,3,9,7,3,4,2,1]))
```

```
1
```

In [10]:

```python
# Function to identify the element with second highest frequency
# if there are many such elements, return smallest number
# [1,2,3,2,1,4,4]-->1

def secondHighest(li):
    unique={}
    for n in li:
        if n in unique:
            unique[n]+=1
        else:
            unique[n]=1
    print(unique)

    freq=unique.values()
    uniquemax=[]
    for i in  freq:
        if i not in uniquemax:
            uniquemax.append(i)
    uniquemax=sorted(uniquemax,reverse=True)


    secondmax=uniquemax[1]

    secondmaxkeys=[]

    for item in unique.items():
        if item[1]==secondmax:
            secondmaxkeys.append(item[0])
    return min(secondmaxkeys)
print(secondHighest([1,2,3,2,1,4,4,4]))
```

```
{1: 2, 2: 2, 3: 1, 4: 3}
1
```

In [2]:

```
# Function to identify Kth highest frequency element in the list
# If there are many such elements,return the smallest number
#[9,8,7,6,5,2,3,4,9,6,7,7,7,6,7,6]-> k=4->2


def kHighestfreq(li,k):
    unique={}
    for n in li:
        if n in unique:
            unique[n]+=1
        else:
            unique[n]=1

    freq=unique.values()
    uniquemax=[]
    for i in  freq:
        if i not in uniquemax:
            uniquemax.append(i)
    uniquemax=sorted(uniquemax,reverse=True)


    if len(uniquemax)>=k:
        kmax=uniquemax[k-1]
        kmaxkeys=[]
        for item in unique.items():
            if item[1]==kmax:
                kmaxkeys.append(item[0])
        return min(kmaxkeys)
    return -1
filename='DataFiles/largest.txt'
with open(filename,'r') as f:
    t=int(f.readline())
    for i in range(t):
        li=f.readline()
        k=int(f.readline())
        print(kHighestfreq(li,k))

```

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

In [13]:

```python
def kLowestfreq(li):
    k=int(input())
    unique={}
    for n in li:
        if n in unique:
            unique[n]+=1
        else:
            unique[n]=1
    print(unique)

    freq=unique.values()
    uniquelowest=[]
    for i in  freq:
        if i not in uniquelowest:
            uniquelowest.append(i)
    uniquelowest=sorted(uniquelowest)
    print(uniquelowest)


    if len(uniquelowest)>=k:
        klowest=uniquelowest[k-1]
        klowestkeys=[]
        for item in unique.items():
            if item[1]==klowest:
                klowestkeys.append(item[0])
        return min(klowestkeys)
    return -1

print(kLowestfreq([9,8,7,6,5,2,3,4,9,6,7,7,7,6,7,6]))
```

```
4
{9: 2, 8: 1, 7: 5, 6: 4, 5: 1, 2: 1, 3: 1, 4: 1}
[1, 2, 4, 5]
7
```

In [ ]:

```python
#
```