

Task 12

Registration ID : SIRSS1038

Leela Satya Kartheek Raja

Q.1) Given an array of size n , find all elements in array that appear more than n/k times. For example: if the input array is {3, 1, 2, 2, 1, 2, 3, 3} and k is 4, then the output should be [2, 3]. Note that size of array is 8 (or $n = 8$), so we need to find all elements that appear more than 2 (or $8/4$) times. There are two elements that appear more than two times, 2 and 3.

In [1]:

```
def Mbyk(arr, n, k) :
    x = n // k

    freq = {}

    for i in range(n):

        if arr[i] in freq:
            freq[arr[i]] += 1

        else:
            freq[arr[i]] = 1

    for i in freq :

        if (freq[i] > x) :
            print(i)
```

In [2]:

```
arr = [ 3, 1, 2, 2, 1, 2, 3, 3 ]
n = len(arr)
k = 4
Mbyk(arr, n, k)
```

3
2

Q2. Given an array of size n and a range $[a, b]$. The task is to partition the array around the range such that array is divided into three parts. 1) All elements smaller than a come first. 2) All elements in range a to b come next. 3) All elements greater than b appear in the end. The individual elements of three sets can appear in any order. You are required to return the modified array.

In [3]:

```
def findElement (a, n):

    # Base case
    if (n == 1 or n == 2):
        return -1

    element, maxx, bit = a[0], a[0], -1
    check = 0
    idx = -1

    i = 1
    while (i < (n - 1)):

        if (a[i] < maxx and i < (n - 1)):
            i += 1
            bit = 0
```

```

else:
    if (a[i] >= maxx):
        element = a[i]
        idx = i
        check = 1
        maxx = a[i]

    if (check == 1):
        i += 1

    bit = 1
    while (a[i] >= element and i < (n - 1)):
        if (a[i] > maxx):
            maxx = a[i]

        i += 1

    check = 0
if (element <= a[n - 1] and bit == 1):
    return idx
else:
    return -1

```

In [4]:

```

if __name__ == '__main__':

    arr = [ 1,2,3,4,5 ]
    n = len(arr)

    print(findElement(arr, n))

```

1

Q3. Given an array of integers (A[]) and a number x, find the smallest subarray with sum greater than the given value. Note: The answer always exists. It is guaranteed that x doesn't exceed the summation of a[i] (from 1 to N).

In [5]:

```

def smallestSubWithSum(arr, n, x):

    curr_sum = 0
    min_len = n + 1

    start = 0
    end = 0
    while (end < n):

        while (curr_sum <= x and end < n):
            curr_sum += arr[end]
            end += 1

        while (curr_sum > x and start < n):

            if (end - start < min_len):
                min_len = end - start

            curr_sum -= arr[start]
            start += 1

    return min_len

```

In [6]:

```

arr1 = [1, 4, 45, 6, 0, 19]
x = 51
n1 = len(arr1)
res1 = smallestSubWithSum(arr1, n1, x)
print("Not possible") if (res1 == n1 + 1) else print(res1)

```

Q4. Estimating the cost Problem Statement – We want to estimate the cost of painting a property. Interior wall painting cost is Rs.18 per sq.ft and exterior wall painting cost is Rs.12 per sq.ft.

Take input as

- 1. Number of Interior walls**
- 2. Number of Exterior walls**
- 3. Surface Area of each Interior 4. Wall in units of square feet Surface Area of each exterior wall in units of square feet.**

Note – If a user enters zero as the number of walls then skip surface area values as user may don't want to paint that wall. Calculate and display the total cost of painting the property

In [7]:

```
iw = int(input())
ew = int(input())

if iw:
    int_walls = []
    for i in range(iw):
        int_walls.append(float(input()))

if ew:
    ext_walls = []
    for i in range(ew):
        ext_walls.append(float(input()))
if ew < 0 or iw < 0:
    print("Invalid Input")
    exit()

if ew and iw:
    print("Total estimated Cost : ", (sum(int_walls)*18+sum(ext_walls)*12), "INR")
else:
    if ew:
        print("Total estimated Cost : ", sum(ext_walls)*12, "INR")
    elif iw:
        print("Total estimated Cost : ", sum(int_walls)*18, "INR")
    else:
        print("Total estimated Cost : 0.0 INR")
```

```
6
3
13.5
17.5
15.5
15.5
10.0
12.5
10.236
15.258
10.26
Total estimated Cost : 1950.048 INR
```

In []: