# Experiment No – 1.3

**Student Name: Abhishek Kumar Gupta** **UID: 18BCS2219**
**Branch: BE-CSE** **Section/Group: CSE-11 'B'**
**Semester: 5** **Date of Performance: 08/08/2020**
**Subject Name: DAA LAB** **Subject Code: CSP-309**

## 1. Aim/Overview of the practical:

**Find the frequency of each element of an array with the complexity of O(n).**

## 2. Task to be done:

**Counting frequencies of array elements**

Given an array which may contain duplicates, print all elements and their frequencies.

**Examples:**

Input : arr[] = {10, 20, 20, 10, 10, 20, 5, 20}

Output : 10 3

20 4

5 1

**3. Algorithm/Flowchart** (For programming based labs)**:**

Step-1: **Input the number of elements of the array.**
Step-2: **Input the array elements.**
Step-3: **for(i=0;i<n;i++)**

```
{
        a[i] = a[i]-1;
}
```

Step-4: **for(i=0;i<n;i++)**

```
{
        id = (a[i]%n);
        a[id] = a[id] + n;
}
```

Step-5: **for(i=0;i<n;i++)**

```
{
        count = a[i]/n;
        a[i] = a[i]%n+1;
        cout<<i+1<<" appears "<<count<<" times."<<"\n";
}
```

Step-6: **Finish.**

**4. Steps for experiment/practical:**

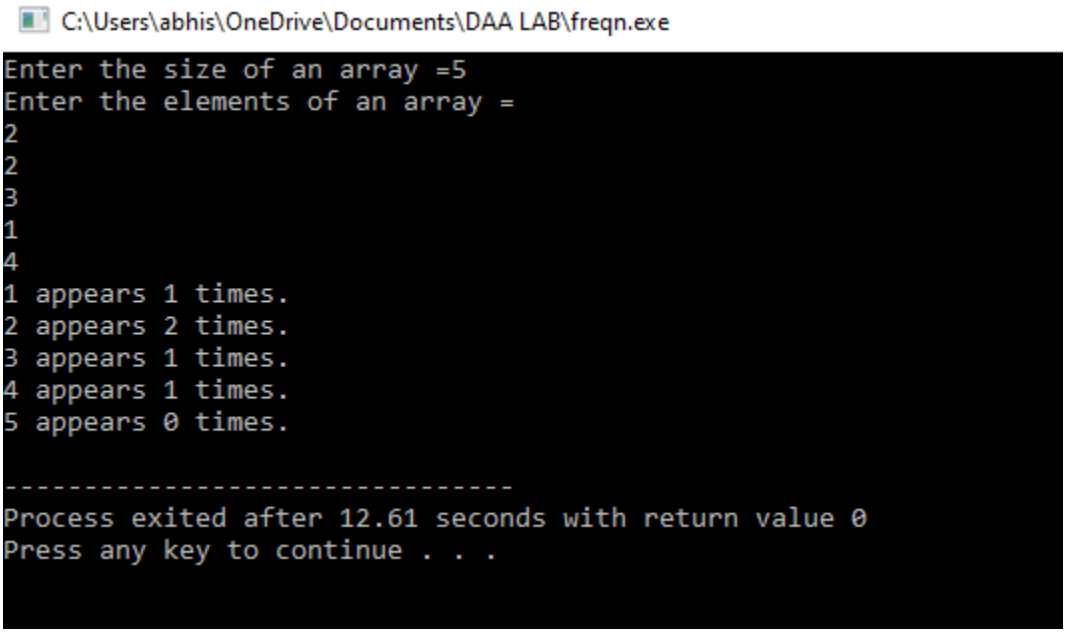
```
#include<iostream>
using namespace std;

void countFreq(int a[],int n)
{
        int i,id,count;
        for(i=0;i<n;i++)
```

```
        {
                a[i] = a[i]-1;
        }
        for(i=0;i<n;i++)
        {
                id = (a[i]%n);
                a[id] = a[id] + n;
        }
        for(i=0;i<n;i++)
        {
                count = a[i]/n;
                a[i] = a[i]%n+1;
                cout<<i+1<<" appears "<<count<<" times."<<"\n";
        }
}
int main()
{
        int a[100],i,n;
        cout<<"Enter the size of an array =";
        cin>>n;
        cout<<"Enter the elements of an array =\n";
        for(i=0;i<n;i++)
        {
                cin>>a[i];
        }
        countFreq(a,n);
}
```

**5. Observations/Discussions:**

Time Complexity of this algorithm**: O(n)**

## 6. Result/Output/Writing Summary:




## Learning outcomes (What I have learnt):

1. **I have learnt the to find the frequency of a series of number.**
2. **The complexity of this program is O(n).**
3. 
4. 
5. 

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
|---|---|---|---|
| 1. | | | |
| 2. | | | |
| 3. | | | |
| | | | |