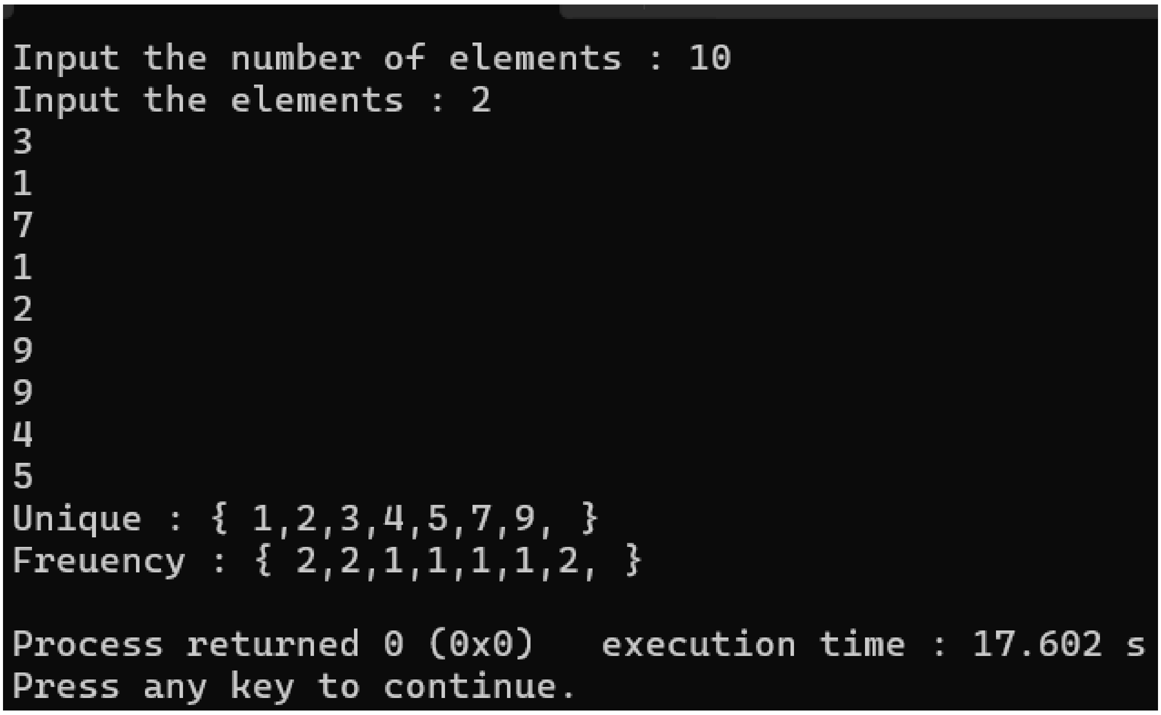
15B17CI371 – Data Structures Lab
ODD 2024
Week 4-LAB B
Practice Lab

1. Write a program to count the frequency of each element in a given array. Test Case:

```
#include <iostream>
using namespace std;
int getMax(int *arr,int n)
{
   int max=arr[0];
   for(int i=0;i<n;i++)
      if(arr[i]>max)
      max=arr[i];
   return max;
}
int main()
{
   int n;
   cout<<"Input the number of elements : ";
   cin>>n;
   int *arr=new int[n];
   cout<<"Input the elements : ";
   for(int i=0;i<n;i++)
      cin>>arr[i];
   int s=getMax(arr,n)+1;
   int count[s]={0};
   for(int i=0;i<n;i++)
      count[arr[i]]++;
   cout<<"Unique : { ";
   for(int i=0;i<s;i++)
      if(count[i]>0)
         cout<<i<<",";
   cout<<" }"<<endl<<"Freuency : { ";
   for(int i=0;i<s;i++)
      if(count[i]>0)
         cout<<count[i]<<",";
   cout<<" }"<<endl;
}
```

Output :



```
Input the number of elements : 10
Input the elements : 2
3
1
7
1
2
9
9
4
5
Unique : { 1,2,3,4,5,7,9, }
Freuency : { 2,2,1,1,1,1,2, }

Process returned 0 (0x0)   execution time : 17.602 s
Press any key to continue.
```

2. Given a sorted array of n elements,write a function to search for a given target value using the Jump Search algorithm. The Jump Search algorithm works by dividing the array into blocks of a fixed size,jumping ahead by these block sizes,and then performing a linear search within the block where the target might be present.

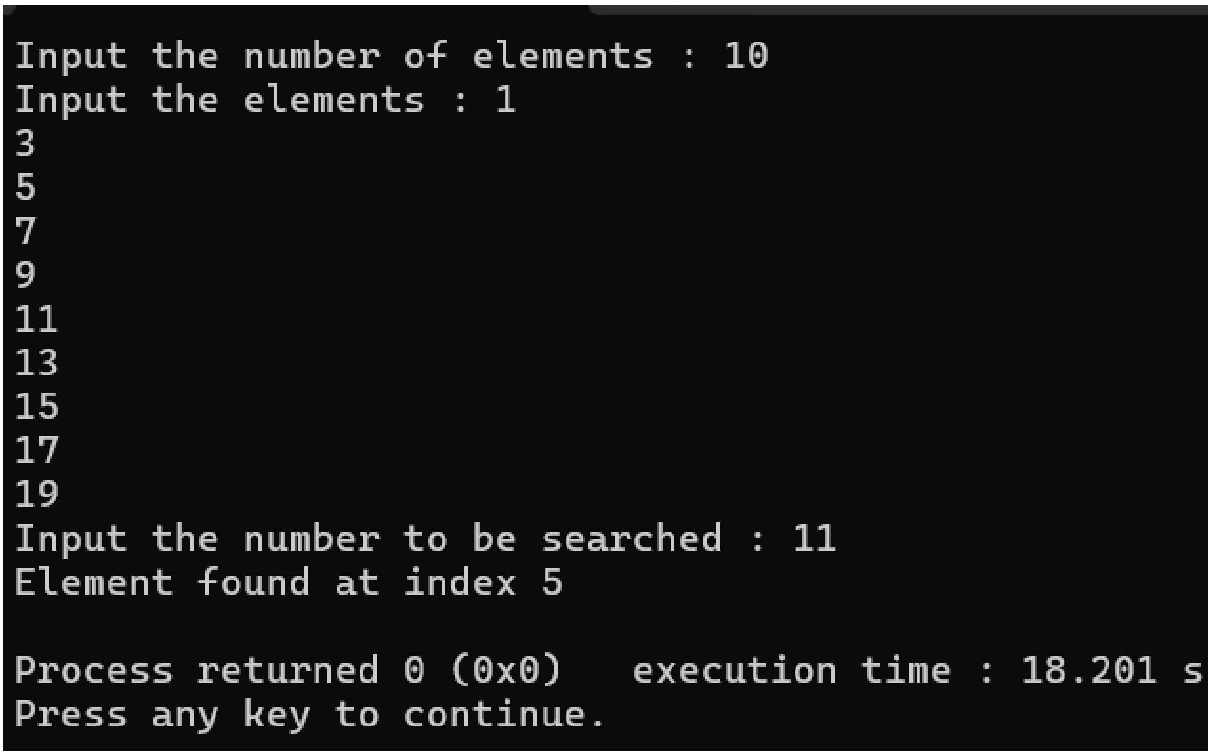
```
#include <iostream>
#include <cmath>
using namespace std;
int getMin(int *arr,int n)
{
   int min=arr[0];
   for(int i=0;i<n;i++)
      if(arr[i]>min)
         min=arr[i];
   return min;
}
int jumpSearch(int *arr,int n,int key)
{
   int step=sqrt(n);
   int prev=0;
   while(arr[min(step,n)-1]<key)
   {
      prev=step;
      step+=sqrt(n);
      if(prev>=n)
         return -1;
   }

   while(arr[prev]<key)
   {
      prev++;
      if(prev==min(step,n))
         return -1;
   }

   if(arr[prev]==key)
      return prev;
```

```
    return -1;
}
int main()
{
    int n;
    cout<<"Input the number of elements : ";
    cin>>n;
    int *arr=new int[n];
    cout<<"Input the elements : ";
    for(int i=0;i<n;i++)
        cin>>arr[i];
    int key;
    cout<<"Input the number to be searched : ";
    cin>>key;
    int index=jumpSearch(arr,n,key);
    if(index != -1)
        cout<<"Element found at index "<<index<<endl;
    else
        cout<<"Element not found"<<endl;
}
```

Output :



```
Input the number of elements : 10
Input the elements : 1
3
5
7
9
11
13
15
17
19
Input the number to be searched : 11
Element found at index 5

Process returned 0 (0x0)   execution time : 18.201 s
Press any key to continue.
```

3. Given an array of integers,sort the array according to the frequency of elements. Elements with higher frequency come first. If two elements have the same frequency, they are sorted by their value.

```
#include <iostream>
#include <cmath>
using namespace std;
struct Element
{
    int value;
    int frequency;
};
```

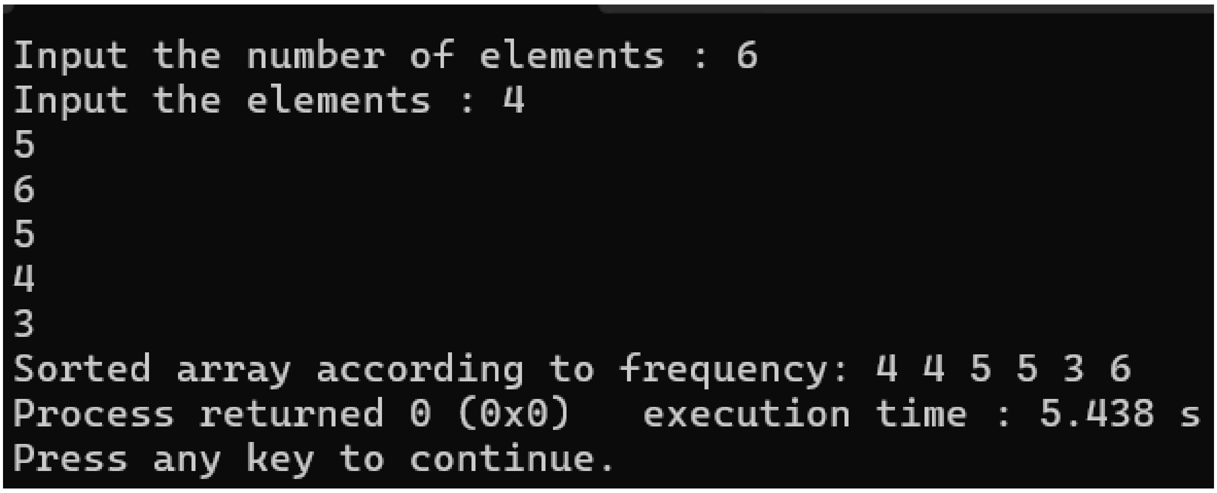
```
void swap(Element &a,Element &b)
{
   Element temp=a;
   a=b;
   b=temp;
}

void sortElements(Element elements[],int n)
{
   for(int i=0;i<n-1;i++)
      for(int j=0;j<n-i-1;j++)
                                                    if(elements[j].frequency<elements[j+1].frequency||
     (elements[j].frequency==elements[j+1].frequency&&elements[j].value>
     elements[j+1].value))
            swap(elements[j],elements[j+1]);
}

void frequencySort(int arr[],int n)
{
   Element elements[n];
   int visited[n];
   for(int i=0;i<n;i++)
   {
      elements[i].value=arr[i];
      elements[i].frequency=1;
      visited[i]=0;
   }
   for(int i=0;i<n;i++)
   {
      if(visited[i]==1)
         continue;
      for(int j=i+1;j<n;j++)
         if(arr[i]==arr[j])
         {
            elements[i].frequency++;
            visited[j]=1;
         }
   }
   sortElements(elements,n);
   int index=0;
   for(int i=0;i<n;i++)
      if(visited[i]==0)
         for(int j=0;j<elements[i].frequency;j++)
            arr[index++]=elements[i].value;
}
int main()
{
   int n;
   cout<<"Input the number of elements : ";
   cin>>n;
   int *arr=new int[n];
   cout<<"Input the elements : ";
   for(int i=0;i<n;i++)
      cin>>arr[i];
   frequencySort(arr,n);
   cout<<"Sorted array according to frequency: ";
   for(int i=0;i<n;i++)
```

```
        cout<<arr[i]<<" ";
}
```

Output :



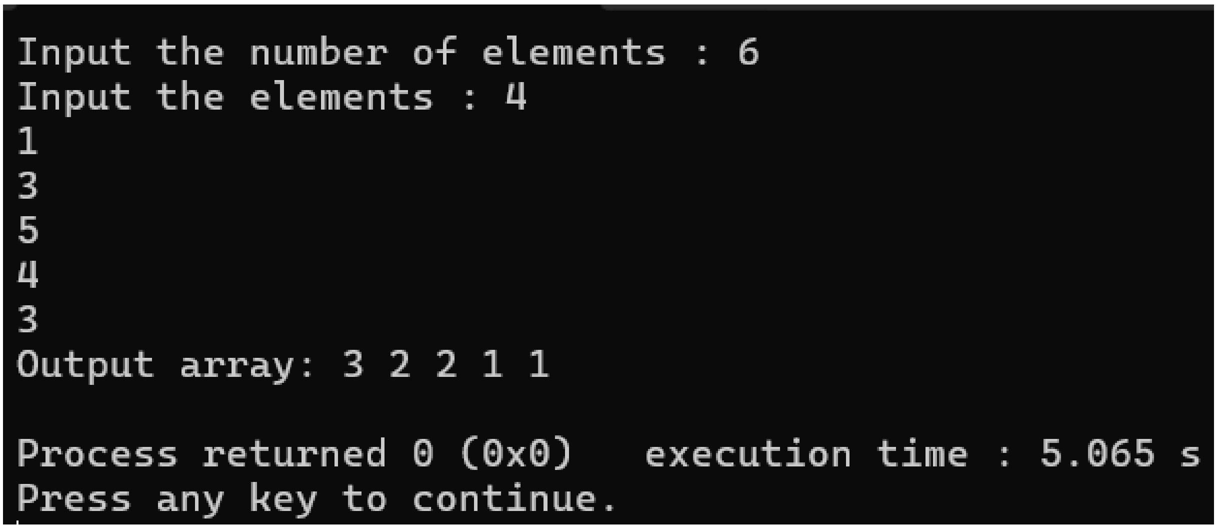
```
Input the number of elements : 6
Input the elements : 4
5
6
5
4
3
Sorted array according to frequency: 4 4 5 5 3 6
Process returned 0 (0x0)   execution time : 5.438 s
Press any key to continue.
```

4. **Given an array of integers,return an array depicting the differences between two successive elements sorted in descending order.**

```
#include <iostream>
#include <cmath>
using namespace std;
void swap(int &a,int &b)
{
   int temp=a;
   a=b;
   b=temp;
}

void sortDescending(int arr[],int n)
{
   for(int i=0;i<n-1;i++)
      for(int j=0;j<n-i-1;j++)
         if(arr[j]<arr[j+1])
            swap(arr[j],arr[j+1]);
}
void computeDifferences(int arr[],int n,int diff[])
{
   for(int i=0;i<n-1;i++)
      diff[i]=abs(arr[i+1]-arr[i]);
}
void differenceArray(int arr[],int n)
{
   int diff[n-1];
   computeDifferences(arr,n,diff);
   sortDescending(diff,n-1);
   cout<<"Output array: ";
   for(int i=0;i<n-1;i++)
      cout<<diff[i]<<" ";
   cout<<endl;
}
int main()
{
    int n;
    cout<<"Input the number of elements : ";
    cin>>n;
    int *arr=new int[n];
    cout<<"Input the elements : ";
    for(int i=0;i<n;i++)
        cin>>arr[i];
    differenceArray(arr,n);
}
```

Output :



```
Input the number of elements : 6
Input the elements : 4
1
3
5
4
3
Output array: 3 2 2 1 1

Process returned 0 (0x0)   execution time : 5.065 s
Press any key to continue.
```