RAJSHAHI UNIVERSITY OF ENGINEERING & TECHNOLOGY

SESSIONAL TASK-07

COURSE NAME: SESSIONAL BASED ON CSE-2201
COURSE CODE: CSE-2102

SUBMITTED TO-

BIPRODIP PAL
Assistant Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

SUBMITTED BY-

SRABONTI DEB
Roll- 1803163
Section-C
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology

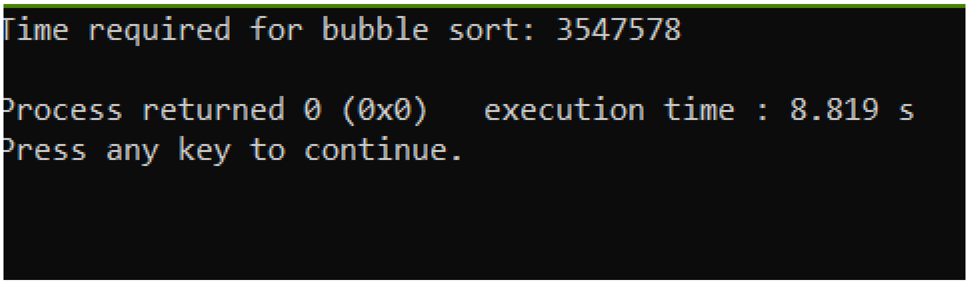
SUBMISSION DATE – 3 JULY, 2021

Problem Statement: Comparison of bubble sort & counting sort algorithms.

Code for Bubble Sort Algorithm:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int cnt,n,i,j,temp,counter;
    long long ar[10000],num;
    ofstream input;
    input.open("input.txt");
    for (cnt=1;cnt<=2000;cnt++)
    {
        n=rand()%2000+1;
        input<<n<<endl;
    }
    input.close();
    ifstream file;
    file.open("input.txt");
    if(file)
    {
        while(file>>num)
        {
            ar[n++]=(num);
        }
    }
    else
        cout<<"file can't open"<<endl;
    int counter1=2;
    for(i=0; i<n; i++)
    {
        counter+=2;
        for(j=1; j<n-i; j++)
        {
            counter+=2;
            if(ar[j]>ar[j+1])
            {
                temp=ar[j];
                ar[j]=ar[j+1];
                ar[j+1]=temp;
                counter=4;
            }
        }
    }
    counter+=2;
    cout<<"Time required for bubble sort: "<<counter<<endl;
}
```

Output:



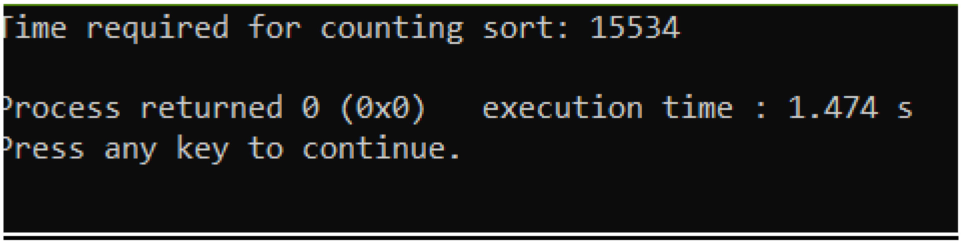
```
Time required for bubble sort: 3547578

Process returned 0 (0x0)   execution time : 8.819 s
Press any key to continue.
```

Code for Counting Sort Algorithm:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int cnt,n,i;
    long long ar[10000],ar1[10000],num;
    ofstream input;
    input.open("input.txt");
    for (cnt=1;cnt<=2000;cnt++)
    {
        n=rand()%2000+1;
        input<<n<<endl;
    }
    input.close();
    ifstream file;
    file.open("input.txt");
     if(file)
    {
        while(file>>num)
        {
            ar[n++]=(num);
        }
    }
    else
        cout<<"file can't open"<<endl;
    long long maximum=*max_element(ar,ar+n),b[maximum+5];
    memset(b,0,sizeof(b));
    memset(ar1,0,sizeof(ar1));
    for(i=0; i<n; i++)
    {
        b[ar[i]]++;
    }
    for(i=1; i<=maximum; i++)
    {
        b[i]=b[i-1]+b[i];
    }
    int counter=1;
    for(i=n-1; i>=0; i--)
    {
        ar1[b[ar[i]] - 1] = ar[i];
        b[ar[i]]--;
        counter+=4;
    }
    counter+=1;
    cout<<"Time required for counting sort: "<<counter<<endl;
}
```

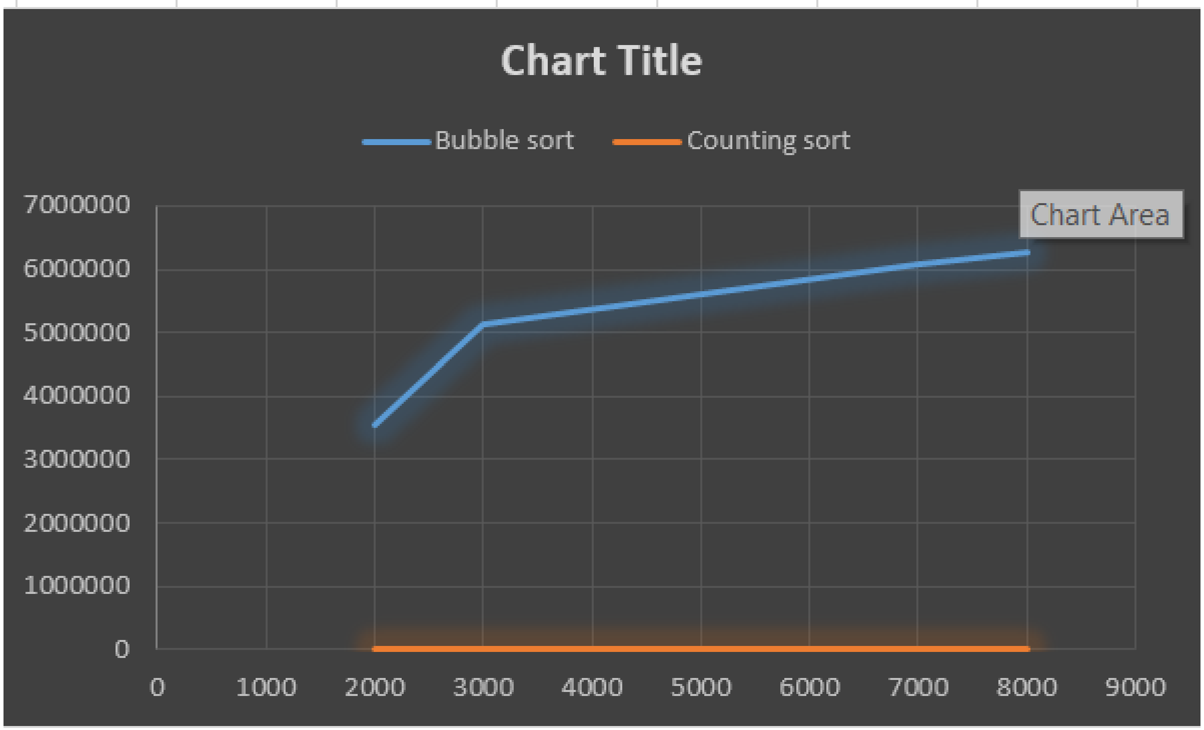
Output:



```
Time required for counting sort: 15534

Process returned 0 (0x0)   execution time : 1.474 s
Press any key to continue.
```

Graph:



Discussion:

We know that the complexity of counting sort is O(n) and bubble sort is O(n^2).That means the complexity of counting sort is less than that of bubble sort.We can prove this by the executed output and graph (shown above).In a nutshell,we can say that counting sort is more efficient than bubble sort.

.