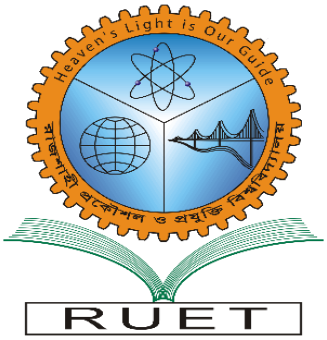


RAJSHAHI UNIVERSITY OF ENGINEERING AND TECHNOLOGY

Course No: CSE 2201

Course Title: Sessional Based on CSE-2201

Submitted to:
Biprodip Pal

Assistant Professor ,
Department of Computer Science and Engineering
Rajshahi University of Engineering and Technology

Submitted by:
Md Al Amin Tokder Shoukhin ,

Roll: 1803078, **Section:** B
Department of Computer Science and Engineering,
Rajshahi University of Engineering and Technology

Problem Name: Sorting in linear time (Counting sort)

Code(Bubble Sort):

```
package algorithm;

import java.io.File;
import java.util.Scanner;

public class _1803078{

	public static void main(String[] args){
		try{
			File f=new File("input.txt");
			Scanner ob=new Scanner(f);
			int n=ob.nextInt();
			int a[]=new int[n];
			for(int i=0;i<n;i++){
				a[i]=ob.nextInt();
			}
			long t1=System.nanoTime();
			for(int i=0;i<n;i++){
				for(int j=0;j<n-i-1;j++){
					if(a[j]>a[j+1]){
						int temp=a[j];
						a[j]=a[j+1];
						a[j+1]=temp;
					}
				}
			}
			long t2=System.nanoTime();
			System.out.println(t2-t1);

			//for(int i=0;i<n;i++){
			//	System.out.println(a[i]);
			//}

		}catch(Exception e){

		}
```

```
	}
}
```

Code (Counting Sort):

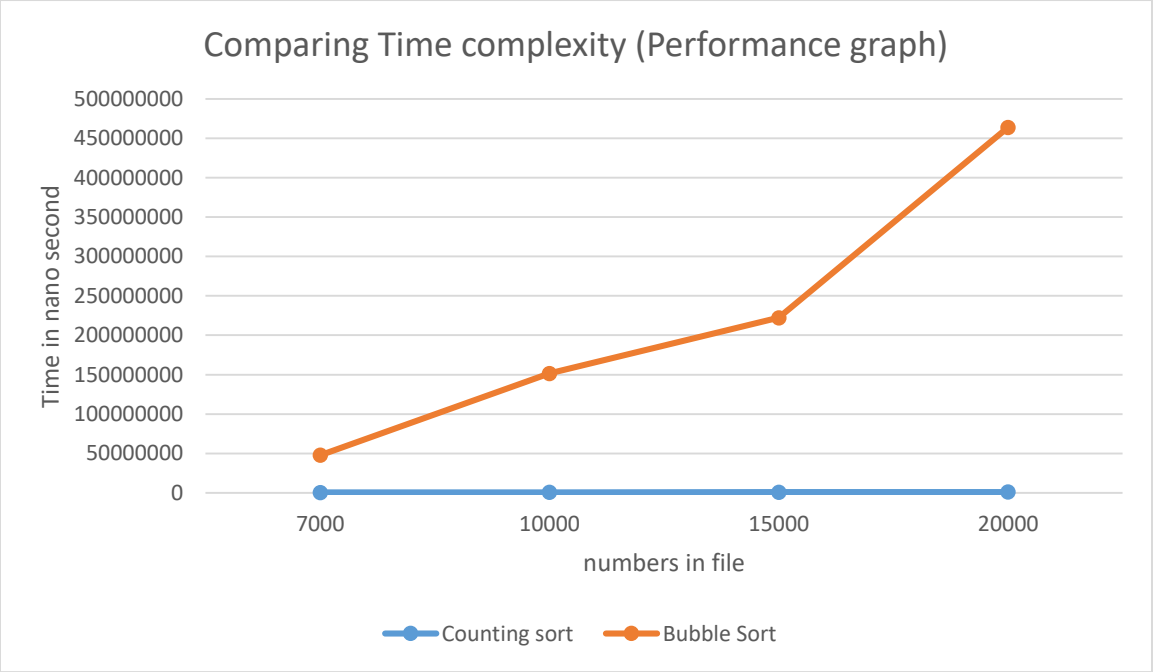
```
package algorithm;

import java.io.File;
import java.util.Scanner;

public class _1803078{

	public static void main(String[] args){
		try{
			File f=new File("input.txt");
			Scanner ob=new Scanner(f);
			int n=ob.nextInt();
			int a[]=new int[n];
			int max=Integer.MIN_VALUE;
			for(int i=0;i<n;i++){
				a[i]=ob.nextInt();
				max=Math.max(max,a[i]);
				//System.out.println(a[i]);
			}

			long t1=System.nanoTime();

			int b[]=new int[max+1];
			for(int i=0;i<a.length;i++){
				b[a[i]]++;
			}
			for(int i=1;i<b.length;i++){
				b[i]=b[i]+b[i-1];
			}

			//for(int i=0;i<b.length;i++){
			//	System.out.println(i+" "+b[i]);
			//}

			int sorted_array[]=new int[b[b.length-1]];
			for(int i=0;i<a.length;i++){
```

```
                        int pos=b[a[i]];
                        sorted_array[pos-1]=a[i];
                        b[a[i]]--;
                    }

                    long t2=System.nanoTime();
                    System.out.println(t2-t1);


                    //for(int i=0;i<sorted_array.length;i++){
                    //      System.out.println(sorted_array[i]);
                    //}

                }catch(Exception e){

                }


        }
}
```

Time Count (Nano second):

Numbers in File (data)	Counting Sort (nano sec.)	Bubble Sort (nano sec.)
7000	352100	47732500
10000	569400	151442500
15000	764200	222330600
20000	1023700	463981500

Performance analysis :



Comment on Findings:

From the performance graph I see that bubble sort taken more and more time rather than counting sort.

when my input data size is 7000 then , counting sort takes 352100 nano sec and bubble sort takes 47732500 nano sec.

similarly ,

when my input data size is 10000 then , counting sort takes 569400 nano sec. and bubble sort takes 151442500 nano sec. etc.

It seems that counting sort works in linear time complexity and bubble sort work Quadratic complexity.

Counting sort is better than bubble sort.