

# DESIGN AND ANALYSIS OF ALGORITHM

**K S PRABHATH**

**19BCE7564**

**QUESTION: DIVIDE AND CONQUER ONLINE BET**

**CODE:**

```
import java.util.*;
public class p1{
    static Scanner in=new Scanner(System.in);
    static ArrayList<Integer> sVal, count;
    private static void calculate(int[] point,
    ArrayList<Integer> ar) {
        for(int i=0;i<point.length;i++){
            if(point[i]<=ar.get(1)&&point[i]>=ar.get(0)){
                int temp=count.get(i);
                count.set(i, ++temp);
            }
        }
    }
    private static int[] enterValues(int[]
    a, int p) {
        for(int i=0;i<p;i++)
            a[i]=in.nextInt();
        return a;
    }
    public static void main(String[] args) {
        System.out.println("No.of Segments: "); int s=in.nextInt();
        System.out.println("No.of Points: ");
        int p=in.nextInt();
```

```

int[] point=new int[p];
int[] point=new int[p];
System.out.println("Enter the values in point array: ");
point=enterValues(point,p);
count=new ArrayList<Integer>();
for(int i=0;i<p;i++){
count.add(0);
}
sVal=new ArrayList<Integer>(s);
for(int i=0;i<s;i++)
{
System.out.println("Enter values for the segment: ");
for(int j=0;j<2;j++)
sVal.add(in.nextInt());
calculate(point,sVal);
sVal.clear();
}
for(int i=0;i<p;i++)
System.out.print(count.get(i)+" ");
}
}

```

**OUTPUT:**

The screenshot shows the JDoodle online Java compiler interface. The browser address bar displays `jdoodle.com/online-java-compiler/`. The code editor contains the following Java code:

```
43 for(int i=0;i<p;i++)
44 System.out.print(count.get(i)+" ");
45 }
46 }
47
48
```

Below the code editor, the 'Execute Mode, Version, Inputs & Arguments' section is visible. It includes a dropdown menu for the JDK version (JDK 11.0.4), an 'Interactive' checkbox, and a 'Stdin Inputs' text area. The 'CommandLine Arguments' section is also present. A blue 'Execute' button is located to the right of the command line arguments.

The 'Result' section shows the output of the program:

```
CPU Time: sec(s), Memory: kilobyte(s)

No.of Segments:
No.of Points:
Enter the values in point array:
Enter values for the segment:
Enter values for the segment:
2 1
```

## ANALYSIS:

Considering the no.of segments to be  $s$ ; In the for loop in main-- the computational time would be  $O(2s)$  which is equivalent to  $O(s)$