

## **Competitive Programming**

### **Week3.6**

#### **Assignment-3(Wednesday)**

##### **Segment Tree with Point Updates**

###### **Code:**

```
import java.util.*;

class Main

{
    static long[] seg;

    static int n;

    static void build(int idx,int l,int r,int[] arr)

    {
        if(l==r)

        {
            seg[idx]=arr[l];

            return;
        }

        int mid=(l+r)/2;

        build(2*idx+1,l,mid,arr);

        build(2*idx+2,mid+1,r,arr);

        seg[idx]=seg[2*idx+1]+seg[2*idx+2];
    }

    static void update(int idx,int l,int r,int pos,int val)

    {
        if(l==r)

        {
            seg[idx]=val;
        }
    }
}
```

```

    return;
}

int mid=(l+r)/2;

if(pos<=mid)

    update(2*idx+1,l,mid,pos,val);

else

    update(2*idx+2,mid+1,r,pos,val);

seg[idx]=seg[2*idx+1]+seg[2*idx+2];

}

static long query(int idx,int l,int r,int ql,int qr)

{

if(qr<l||ql>r)

    return 0;

if(ql<=l&&r<=qr)

    return seg[idx];

int mid=(l+r)/2;

return query(2*idx+1,l,mid,ql,qr)

    + query(2*idx+2,mid+1,r,ql,qr);

}

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

int T=sc.nextInt();

while(T-->0)

{

n=sc.nextInt();

```

```
int[] arr=new int[n];
for(int i=0;i<n;i++)
    arr[i]=sc.nextInt();

seg=new long[4*n];
build(0,0,n-1,arr);

int Q=sc.nextInt();
while(Q-->0)
{
    int type=sc.nextInt();
    if(type==1)
    {
        int i=sc.nextInt();
        int x=sc.nextInt();
        update(0,0,n-1,i,x);
    }
    else
    {
        int L=sc.nextInt();
        int R=sc.nextInt();
        System.out.println("ANS: "+query(0,0,n-1,L,R));
    }
}
}
```

The screenshot shows the OnlineGDB Java IDE interface. The code editor displays a Java file named Main.java. The code implements a segment tree for range queries and updates. It includes methods for building the tree, updating values, and querying ranges. The code uses a recursive approach to build the tree and handle queries. The IDE has a toolbar with Run, Debug, Stop, Share, Save, and Beautify buttons. The Language dropdown is set to Java. The status bar at the bottom shows the date (09-02-2026) and time (11:54). The system tray indicates it's 22°C and sunny.

```
1 import java.util.*;
2 class Main
3 {
4     static long[] seg;
5     static int n;
6     static void build(int idx,int l,int r,int[] arr)
7     {
8         if(l==r)
9         {
10             seg[idx]=arr[1];
11             return;
12         }
13         int mid=(l+r)/2;
14         build(2*idx+1,l,mid,arr);
15         build(2*idx+2,mid+1,r,arr);
16         seg[idx]=seg[2*idx+1]+seg[2*idx+2];
17     }
18     static void update(int idx,int l,int r,int pos,int val)
19     {
20         if(l==r)
21         {
22             seg[idx]=val;
23             return;
24         }
25         int mid=(l+r)/2;
26         if(pos<mid)
27             update(2*idx+1,l,mid,pos,val);
28         else
29             update(2*idx+2,mid+1,r,pos,val);
30         seg[idx]=seg[2*idx+1]+seg[2*idx+2];
31     }
32     static long query(int idx,int l,int r,int ql,int qr)
33     {
34         if(ql>l||qr>r)
35             return 0;
36         if(ql==l&&qr==r)
37             return seg[idx];
38         int mid=(l+r)/2;
39         return query(2*idx+1,l,mid,ql,qr)+query(2*idx+2,mid+1,r,ql,qr);
40     }
41     public static void main(String[] args)
42     {
43         Scanner sc=new Scanner(System.in);
44         int T=sc.nextInt();
45         while(T-->0)
46         {
47             n=sc.nextInt();
48             int[] arr=new int[n];
49             for(int i=1;i<n;i++)
50                 arr[i]=sc.nextInt();
51             seg=new long[4*n];
52             build(0,0,n-1,arr);
53             int Q=sc.nextInt();
54             while(Q-->0)
55             {
56                 int type=sc.nextInt();
57                 if(type==1)
58                 {
59                     int i=sc.nextInt();
60                     int x=sc.nextInt();
61                     update(0,0,n-1,i,x);
62                 }
63                 else
64                 {
65                     int L=sc.nextInt();
66                     int R=sc.nextInt();
67                     System.out.println("ANS: "+query(0,0,n-1,L,R));
68                 }
69             }
70         }
71     }
72 }
```

This screenshot shows the same Java IDE interface as the first one, but the code in Main.java has been modified. The main loop now reads input from standard input instead of a scanner. It initializes the segment tree, processes updates, and prints the results of range queries directly to the console. The rest of the code remains the same as in the first screenshot.

```
1 (0<1||ql>r)
2     return 0;
3     if(ql==1&&qr==r)
4     {
5         return seg[idx];
6     }
7     int mid=(l+r)/2;
8     return query(2*idx+1,l,mid,ql,qr)+query(2*idx+2,mid+1,r,ql,qr);
9 }
10 public static void main(String[] args)
11 {
12     Scanner sc=new Scanner(System.in);
13     int T=sc.nextInt();
14     while(T-->0)
15     {
16         n=sc.nextInt();
17         int[] arr=new int[n];
18         for(int i=1;i<n;i++)
19             arr[i]=sc.nextInt();
20
21         seg=new long[4*n];
22         build(0,0,n-1,arr);
23
24         int Q=sc.nextInt();
25         while(Q-->0)
26         {
27             int type=sc.nextInt();
28             if(type==1)
29             {
30                 int i=sc.nextInt();
31                 int x=sc.nextInt();
32                 update(0,0,n-1,i,x);
33             }
34             else
35             {
36                 int L=sc.nextInt();
37                 int R=sc.nextInt();
38                 System.out.println("ANS: "+query(0,0,n-1,L,R));
39             }
40         }
41     }
42 }
```

The screenshot shows a Java application running on the OnlineGDB platform. The code in Main.java is as follows:

```
62     if(type==1)
63     {
64         int i=sc.nextInt();
65         int x=sc.nextInt();
66         update(0,0,n-1,i,x);
67     }
68     else
69     {
70         int L=sc.nextInt();
71         int R=sc.nextInt();
72         System.out.println("ANS: "+query(0,0,n-1,L,R));
73     }
74 }
75 }
76 }
77 }
```

The input provided was:

```
1
5
1 3 5 7 9
4
2 1 3
ANS: 15
1 2 10
2 1 3
ANS: 20
2 0 4
ANS: 30
```

The output produced by the program is:

```
...Program finished with exit code 0
```