Competitive Programming

Week5.10_Assignment-3(Wednesday)

Week 5-10 Exploring Applications of Advanced Tree Data

Code:

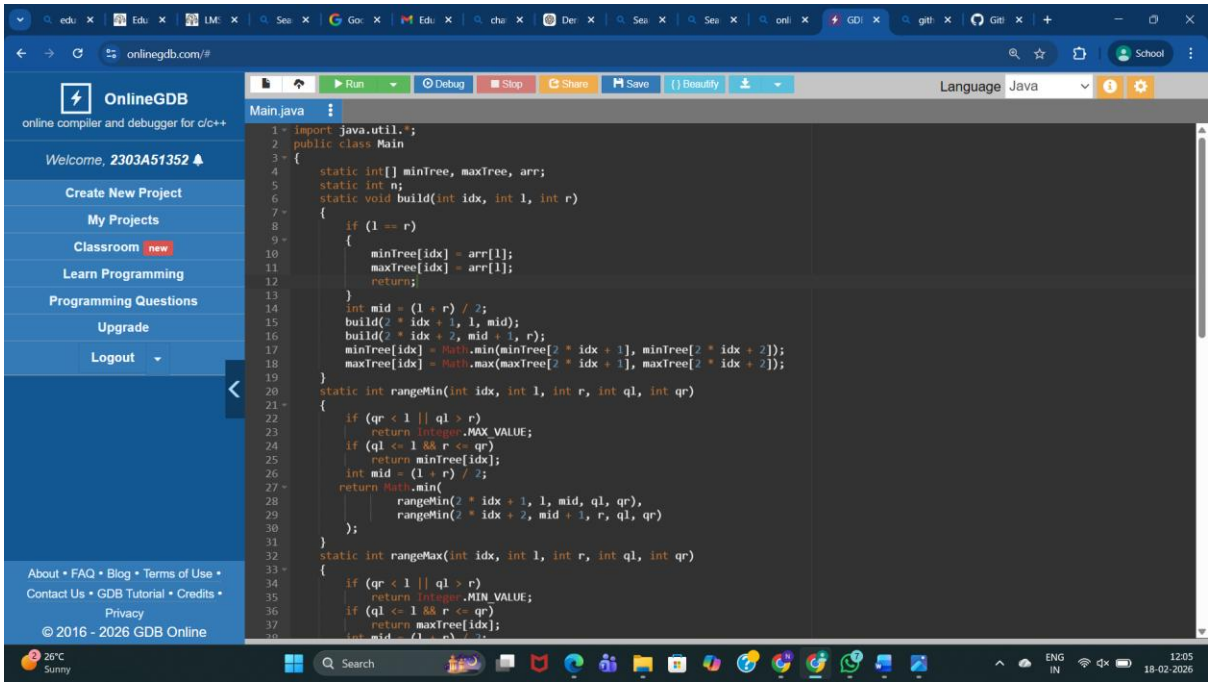
Real-Time Minimum and Maximum Query Using Segment

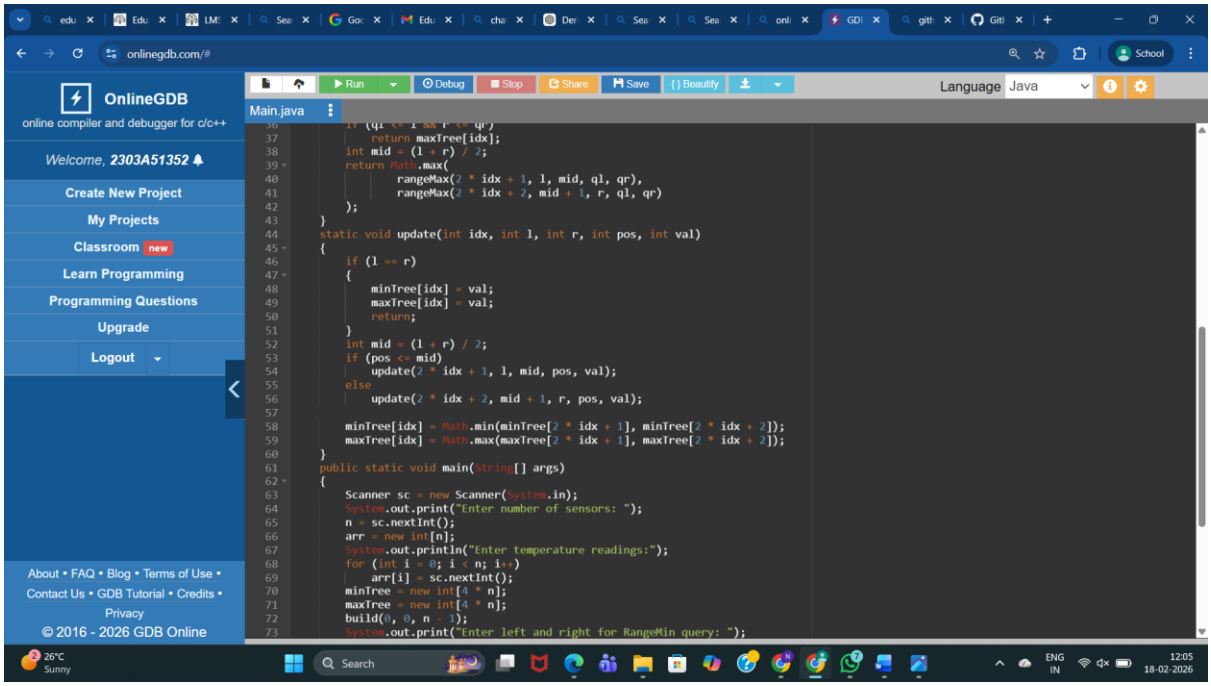
```
import java.util.*;

public class Main

{

  static int[] minTree, maxTree, arr;

  static int n;

  static void build(int idx, int l, int r)

  {

    if (l == r)

    {

      minTree[idx] = arr[l];

      maxTree[idx] = arr[l];

      return;

    }

    int mid = (l + r) / 2;

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

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

    minTree[idx] = Math.min(minTree[2 * idx + 1], minTree[2 * idx + 2]);

    maxTree[idx] = Math.max(maxTree[2 * idx + 1], maxTree[2 * idx + 2]);

  }

  static int rangeMin(int idx, int l, int r, int ql, int qr)

  {

    if (qr < l || ql > r)

      return Integer.MAX_VALUE;

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

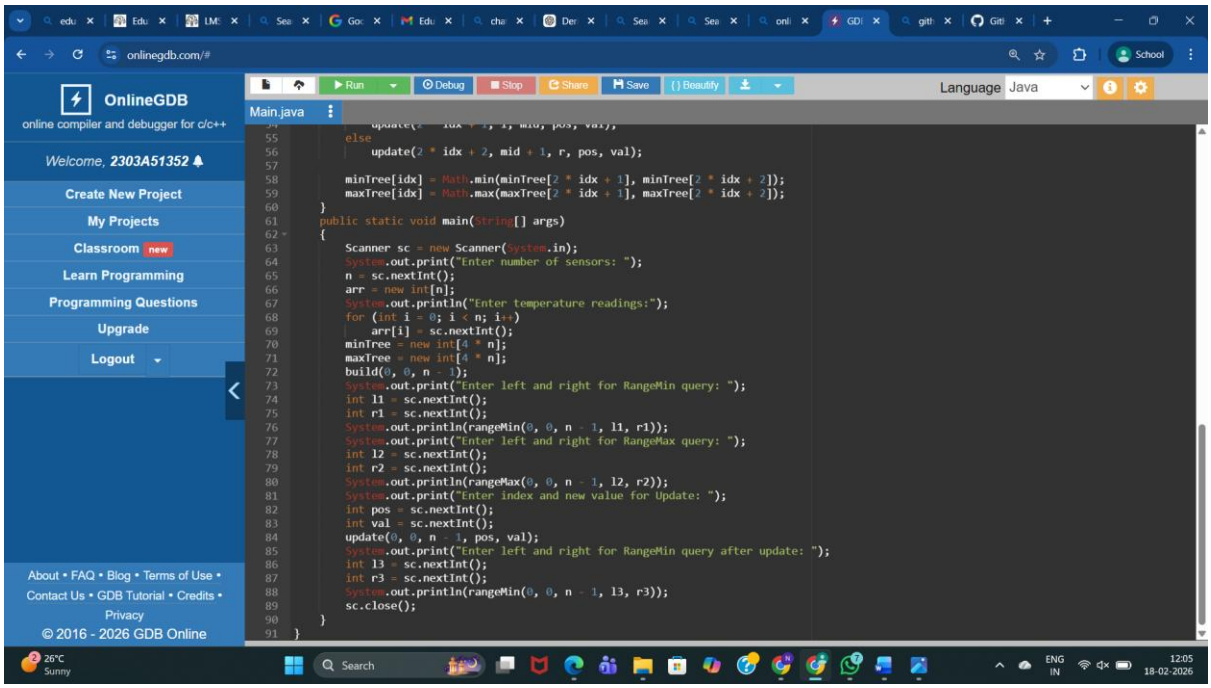
```
        return minTree[idx];
    int mid = (l + r) / 2;
   return Math.min(
          rangeMin(2 * idx + 1, l, mid, ql, qr),
          rangeMin(2 * idx + 2, mid + 1, r, ql, qr)
    );
}
static int rangeMax(int idx, int l, int r, int ql, int qr)
{
    if (qr < l || ql > r)
        return Integer.MIN_VALUE;
    if (ql <= l && r <= qr)
        return maxTree[idx];
    int mid = (l + r) / 2;
    return Math.max(
          rangeMax(2 * idx + 1, l, mid, ql, qr),
          rangeMax(2 * idx + 2, mid + 1, r, ql, qr)
    );
}
static void update(int idx, int l, int r, int pos, int val)
{
    if (l == r)
    {
        minTree[idx] = val;
        maxTree[idx] = val;
        return;
    }
    int mid = (l + r) / 2;
```

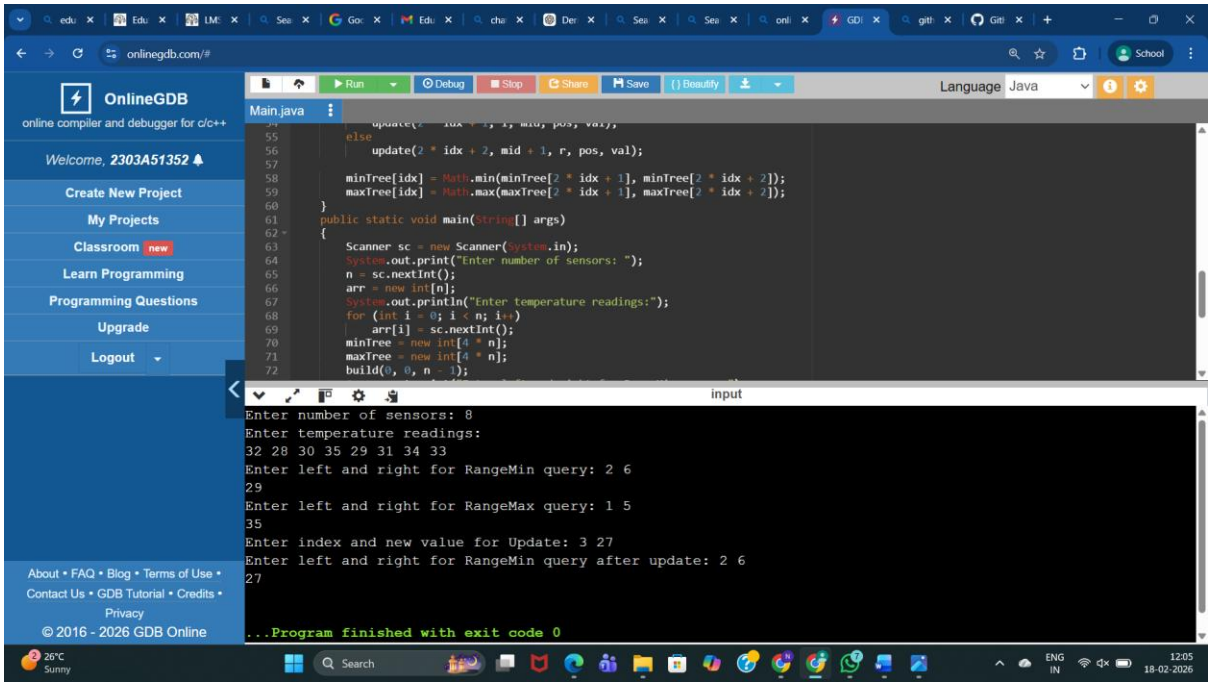
```
    if (pos <= mid)
        update(2 * idx + 1, l, mid, pos, val);
    else
        update(2 * idx + 2, mid + 1, r, pos, val);

    minTree[idx] = Math.min(minTree[2 * idx + 1], minTree[2 * idx + 2]);
    maxTree[idx] = Math.max(maxTree[2 * idx + 1], maxTree[2 * idx + 2]);
}
public static void main(String[] args)
{
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter number of sensors: ");
    n = sc.nextInt();
    arr = new int[n];
    System.out.println("Enter temperature readings:");
    for (int i = 0; i < n; i++)
        arr[i] = sc.nextInt();
    minTree = new int[4 * n];
    maxTree = new int[4 * n];
    build(0, 0, n - 1);
    System.out.print("Enter left and right for RangeMin query: ");
    int l1 = sc.nextInt();
    int r1 = sc.nextInt();
    System.out.println(rangeMin(0, 0, n - 1, l1, r1));
    System.out.print("Enter left and right for RangeMax query: ");
    int l2 = sc.nextInt();
    int r2 = sc.nextInt();
    System.out.println(rangeMax(0, 0, n - 1, l2, r2));
```

```
        System.out.print("Enter index and new value for Update: ");
        int pos = sc.nextInt();
        int val = sc.nextInt();
        update(0, 0, n - 1, pos, val);
        System.out.print("Enter left and right for RangeMin query after update: ");
        int l3 = sc.nextInt();
        int r3 = sc.nextInt();
        System.out.println(rangeMin(0, 0, n - 1, l3, r3));
        sc.close();
    }
}
```




onlinegdb.com/#
OnlineGDB
online compiler and debugger for c/c++
Welcome, 2303A51352
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Upgrade
Logout
Run
Debug
Stop
Share
Save
{ } Beautify
Language Java
Main.java
if (l == r)
return maxTree[idx];
int mid = (l + r) / 2;
return Math.max(
rangeMax(2 * idx + 1, l, mid, ql, qr),
rangeMax(2 * idx + 2, mid + 1, r, ql, qr)
);
}
static void update(int idx, int l, int r, int pos, int val)
{
if (l == r)
{
minTree[idx] = val;
maxTree[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);
minTree[idx] = Math.min(minTree[2 * idx + 1], minTree[2 * idx + 2]);
maxTree[idx] = Math.max(maxTree[2 * idx + 1], maxTree[2 * idx + 2]);
}
public static void main(String[] args)
{
Scanner sc = new Scanner(System.in);
System.out.print("Enter number of sensors: ");
n = sc.nextInt();
arr = new int[n];
System.out.println("Enter temperature readings:");
for (int i = 0; i < n; i++)
arr[i] = sc.nextInt();
minTree = new int[4 * n];
maxTree = new int[4 * n];
build(0, 0, n - 1);
System.out.print("Enter left and right for RangeMin query: ");
About • FAQ • Blog • Terms of Use •
Contact Us • GDB Tutorial • Credits •
Privacy
© 2016 - 2026 GDB Online
26°C
Sunny
Search
ENG
IN
12:05
18-02-2026


onlinegdb.com/#
OnlineGDB
online compiler and debugger for c/c++
Welcome, 2303A51352
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Upgrade
Logout
Run
Debug
Stop
Share
Save
{ } Beautify
Language Java
Main.java
else
update(2 * idx + 2, mid + 1, r, pos, val);
minTree[idx] = Math.min(minTree[2 * idx + 1], minTree[2 * idx + 2]);
maxTree[idx] = Math.max(maxTree[2 * idx + 1], maxTree[2 * idx + 2]);
}
public static void main(String[] args)
{
Scanner sc = new Scanner(System.in);
System.out.print("Enter number of sensors: ");
n = sc.nextInt();
arr = new int[n];
System.out.println("Enter temperature readings:");
for (int i = 0; i < n; i++)
arr[i] = sc.nextInt();
minTree = new int[4 * n];
maxTree = new int[4 * n];
build(0, 0, n - 1);
System.out.print("Enter left and right for RangeMin query: ");
int l1 = sc.nextInt();
int r1 = sc.nextInt();
System.out.println(rangeMin(0, 0, n - 1, l1, r1));
System.out.print("Enter left and right for RangeMax query: ");
int l2 = sc.nextInt();
int r2 = sc.nextInt();
System.out.println(rangeMax(0, 0, n - 1, l2, r2));
System.out.print("Enter index and new value for Update: ");
int pos = sc.nextInt();
int val = sc.nextInt();
update(0, 0, n - 1, pos, val);
System.out.print("Enter left and right for RangeMin query after update: ");
int l3 = sc.nextInt();
int r3 = sc.nextInt();
System.out.println(rangeMin(0, 0, n - 1, l3, r3));
sc.close();
}
}
About • FAQ • Blog • Terms of Use •
Contact Us • GDB Tutorial • Credits •
Privacy
© 2016 - 2026 GDB Online
26°C
Sunny
Search
ENG
IN
12:05
18-02-2026


onlinegdb.com/#
OnlineGDB
online compiler and debugger for c/c++
Welcome, 2303A51352
Create New Project
My Projects
Classroom new
Learn Programming
Programming Questions
Upgrade
Logout
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2026 GDB Online
Run
Debug
Stop
Share
Save
{ } Beautify
Language Java
Main.java
55 else
56 update(2 * idx + 2, mid + 1, r, pos, val);
58 minTree[idx] = Math.min(minTree[2 * idx + 1], minTree[2 * idx + 2]);
59 maxTree[idx] = Math.max(maxTree[2 * idx + 1], maxTree[2 * idx + 2]);
61 public static void main(String[] args)
63 Scanner sc = new Scanner(System.in);
64 System.out.print("Enter number of sensors: ");
65 n = sc.nextInt();
66 arr = new int[n];
67 System.out.println("Enter temperature readings:");
68 for (int i = 0; i < n; i++)
69 arr[i] = sc.nextInt();
70 minTree = new int[4 * n];
71 maxTree = new int[4 * n];
72 build(0, 0, n - 1);
input
Enter number of sensors: 8
Enter temperature readings:
32 28 30 35 29 31 34 33
Enter left and right for RangeMin query: 2 6
29
Enter left and right for RangeMax query: 1 5
35
Enter index and new value for Update: 3 27
Enter left and right for RangeMin query after update: 2 6
27
...Program finished with exit code 0
26°C
Sunny
Search
ENG
IN
12:05
18-02-2026