

Competitive Programming

Week1-Assignment-3(Wednesday)

Assignment 3: Maximum Profit Streak (Divide and Conquer)

Code:

```
import java.util.*;

class Main
{
    static long maxCrossingSum(long[] arr,int left,int mid,int right)
    {
        long sum=0;
        long leftMax=Long.MIN_VALUE;
        for (int i=mid;i>=left;i--)
        {
            sum=sum+arr[i];
            leftMax = Math.max(leftMax, sum);
        }
        sum = 0;
        long rightMax=Long.MIN_VALUE;
        for(int i=mid+1;i<=right; i++)
        {
            sum += arr[i];
            rightMax = Math.max(rightMax, sum);
        }
        return leftMax + rightMax;
    }

    static long maxSubarraySum(long[] arr,int left,int right)
```

```

{
    if (left == right)
    {
        return arr[left];
    }

    int mid = (left + right) / 2;

    long leftSum = maxSubarraySum(arr, left, mid);
    long rightSum = maxSubarraySum(arr, mid + 1, right);
    long crossSum = maxCrossingSum(arr, left, mid, right);
    return Math.max(Math.max(leftSum, rightSum), crossSum);
}

public static void main(String[] args)
{
    Scanner sc = new Scanner(System.in);

    int T=sc.nextInt();
    while (T-- > 0) {
        int N=sc.nextInt();

        long[] arr = new long[N];
        for (int i=0;i<N;i++)
        {
            arr[i] = sc.nextLong();
        }

        System.out.println(maxSubarraySum(arr, 0, N - 1));
    }

    sc.close();
}
}

```

online gdb x DivideAndConquer x chatgpt - Se x Maximum Pr x convert the x Java to PDF x github - Se x NandithaM x Search - Mic x +

onlinegdb.com/edit/BY1OetpSm#

OnlineGDB
online compiler and debugger for c/c++

Welcome, 2303A51352

DivideAndConquer

Create New Project

My Projects

Classroom **new**

Learn Programming

Programming Questions

Upgrade

Logout

auth0

About • FAQ • Blog • Terms of Use •
Contact Us • GDB Tutorial • Credits •
Privacy

© 2016 - 2026 GDB Online

Run Debug Stop Share Save Beautify

Language Java

```
1 import java.util.*;
2 class Main
3 {
4     static long maxCrossingSum(long[] arr,int left,int mid,int right)
5     {
6         long sum=0;
7         long leftMax=Long.MIN_VALUE;
8         for (int i=mid;i>=left;i--)
9         {
10             sum+=arr[i];
11             leftMax = Math.max(leftMax, sum);
12         }
13         sum = 0;
14         long rightMax=Long.MIN_VALUE;
15         for (int i=mid+1;i<=right; i++)
16         {
17             sum += arr[i];
18             rightMax = Math.max(rightMax, sum);
19         }
20         return leftMax + rightMax;
21     }
22     static long maxSubarraySum(long[] arr,int left,int right)
23     {
24         if (left == right)
25         {
26             return arr[left];
27         }
28         int mid = (left + right) / 2;
29         long leftSum = maxSubarraySum(arr, left, mid);
30         long rightSum = maxSubarraySum(arr, mid + 1, right);
31         long crossSum = maxCrossingSum(arr, left, mid, right);
32         return Math.max(Math.max(leftSum, rightSum), crossSum);
33     }
34     public static void main(String[] args)
35     {
36         Scanner sc = new Scanner(System.in);
37         int T=sc.nextInt();
38         while (T-->0) {
```

online gdb x DivideAndConquer x chatgpt - Se x Maximum Pr x convert the x Java to PDF x github - Se x NandithaM x Search - Mic x +

onlinegdb.com/edit/BY1OetpSm#

OnlineGDB
online compiler and debugger for c/c++

Welcome, 2303A51352

DivideAndConquer

Create New Project

My Projects

Classroom **new**

Learn Programming

Programming Questions

Upgrade

Logout

auth0

About • FAQ • Blog • Terms of Use •
Contact Us • GDB Tutorial • Credits •
Privacy

© 2016 - 2026 GDB Online

Run Debug Stop Share Save Beautify

Language Java

```
26         return arr[left];
27     }
28     int mid = (left + right) / 2;
29     long leftSum = maxSubarraySum(arr, left, mid);
30     long rightSum = maxSubarraySum(arr, mid + 1, right);
31     long crossSum = maxCrossingSum(arr, left, mid, right);
32     return Math.max(Math.max(leftSum, rightSum), crossSum);
33 }
34 public static void main(String[] args)
35 {
36     Scanner sc = new Scanner(System.in);
37     int T=sc.nextInt();
38     while (T-->0) {
39         int N=sc.nextInt();
40         long[] arr = new long[N];
41         for (int i=0;i<N;i++)
42         {
43             arr[i] = sc.nextLong();
44         }
45         System.out.println(maxSubarraySum(arr, 0, N - 1));
46     }
47     sc.close();
48 }
49
50
```

input

```
1
9
-2 1 -3 4 -1 2 1 -5 4
6
```

...Program finished with exit code 0
Press ENTER to exit console.