

geeksforgeeks.org/problems/smallest-subarray-with-sum-greater-than-x5651/1

Get 90% Refund Courses Tutorials Practice Jobs

Search... Problem Editorial Submissions Comments

Output Window

Compilation Results Custom Input Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓ Suggest Feedback

Test Cases Passed 1112 / 1112

Attempts : Correct / Total You can see all your attempts in submission tab Accuracy : 100%

Points Scored 1 You can see the score in submission tab

Time Taken 0.38

Java (21) Start Timer

```
1- class Solution {  
2-     public static int smallestSubWithSum(int x, int[] arr) {  
3-         int n = arr.length;  
4-         int minLen = n + 1;  
5-         int sum = 0;  
6-         int start = 0;  
7-  
8-         for (int end = 0; end < n; end++) {  
9-             sum += arr[end];  
10-            while (sum > x) {  
11-                minLen = Math.min(minLen, end - start + 1);  
12-                sum -= arr[start];  
13-                start++;  
14-            }  
15-        }  
16-        return minLen == n + 1 ? 0 : minLen;  
17-    }  
18- }  
19- }  
20- }  
21- }
```

Custom Input Compile & Run Submit

Get 90% Refund

Courses ▾ Tutorials ▾ Practice ▾ Jobs ▾

Search...

Problem Editorial Submissions Comments

Output Window

Compilation Results Custom Input Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓ Suggest Feedback

Test Cases Passed 1112 / 1112 Attempts : Correct / Total 1 / 1 Accuracy : 100%

Points Scored 2 / 2 Time Taken 0.72

Your Total Score: 50 ↑

Java (21) Start Timer

```
1 // User function Template for Java
2
3 class Solution {
4     public int findMinDiff(ArrayList<Integer> arr, int m) {
5         // your code here
6         Collections.sort(arr);
7         int ans = Integer.MAX_VALUE;
8         for(int i = 0; i + m - 1 < arr.size(); i++){
9             ans = Math.min(ans, arr.get(i+m-1) - arr.get(i));
10        }
11
12
13    return ans;
14 }
15 }
```

Solve Next Custom Input Compile & Run Submit