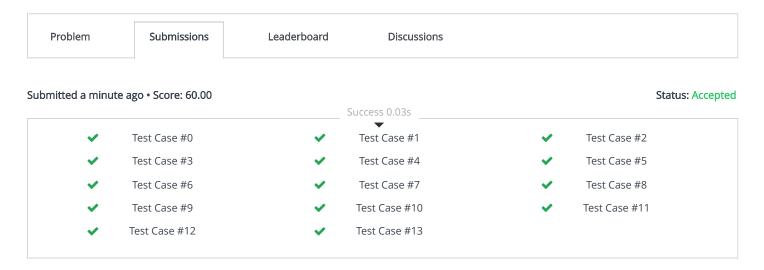
All Contests > In21-CS2023-Lab6 > Game of Two Stacks

Game of Two Stacks



Submitted Code

```
P Open in editor
  nguage: C++
 1 #include <bits/stdc++.h>
 3 using namespace std;
 5 string ltrim(const string &);
 6 string rtrim(const string &);
 7 vector<string> split(const string &);
 8
9 /*
10 * Complete the 'twoStacks' function below.
11 *
12 * The function is expected to return an INTEGER.
13 * The function accepts following parameters:
14 * 1. INTEGER maxSum
15 * 2. INTEGER_ARRAY a
16 * 3. INTEGER_ARRAY b
17
   */
18
19 int twoStacks(int maxSum, vector<int> a, vector<int> b) {
20
       int a_id = 0;
21
       int b_{id} = 0;
       int sum = 0;
22
23
       int count = 0;
24
       while (b_id < (int)b.size() && sum + b[b_id] <= maxSum){</pre>
25
          sum += b[b_id];
26
27
           b_id++;
28
29
       count = b_id;
30
       b_id--;
```

```
31
       while (a_id < (int)a.size() && b_id < (int)b.size()){</pre>
32
           sum += a[a_id];
33
           if (sum > maxSum){
34
               while(b_id >= 0){
35
36
                    sum -= b[b_id];
                    b_id --;
37
                    if (sum <= maxSum){</pre>
38
39
                        break;
40
41
               if (sum > maxSum && b_id < 0){
42
43
                    a_id --;
                    break;
44
               }
45
46
           }
           count = max(a_id+b_id+2, count);
47
48
           a_id ++;
49
       }
50
51
       return count;
52 }
53
54 int main()
55 {
       ofstream fout(getenv("OUTPUT_PATH"));
56
57
58
       string g_temp;
59
       getline(cin, g_temp);
60
61
       int g = stoi(ltrim(rtrim(g_temp)));
62
63
       for (int g_itr = 0; g_itr < g; g_itr++) {</pre>
64
           string first_multiple_input_temp;
65
           getline(cin, first_multiple_input_temp);
66
67
           vector<string> first_multiple_input = split(rtrim(first_multiple_input_temp));
68
           int n = stoi(first_multiple_input[0]);
69
70
71
           int m = stoi(first_multiple_input[1]);
72
           int maxSum = stoi(first_multiple_input[2]);
73
74
75
           string a_temp_temp;
           getline(cin, a_temp_temp);
76
77
78
           vector<string> a_temp = split(rtrim(a_temp_temp));
79
80
           vector<int> a(n);
81
           for (int i = 0; i < n; i++) {
82
               int a_item = stoi(a_temp[i]);
83
84
               a[i] = a_item;
85
86
           }
87
88
           string b_temp_temp;
89
           getline(cin, b_temp_temp);
90
91
           vector<string> b_temp = split(rtrim(b_temp_temp));
92
93
           vector<int> b(m);
94
95
           for (int i = 0; i < m; i++) {
96
               int b_item = stoi(b_temp[i]);
```

```
97
                b[i] = b_item;
98
            }
99
100
            int result = twoStacks(maxSum, a, b);
101
102
103
            fout << result << "\n";</pre>
        }
104
105
        fout.close();
106
107
        return 0;
108
109 }
110
111 string ltrim(const string &str) {
        string s(str);
112
113
114
        s.erase(
115
            s.begin(),
            find_if(s.begin(), s.end(), not1(ptr_fun<int, int>(isspace)))
116
117
        );
118
119
        return s;
120 }
121
122 string rtrim(const string &str) {
123
        string s(str);
124
125
        s.erase(
            find_if(s.rbegin(), s.rend(), not1(ptr_fun<int, int>(isspace))).base(),
126
127
            s.end()
128
        );
129
130
        return s;
131 }
132
133 vector<string> split(const string &str) {
134
        vector<string> tokens;
135
136
        string::size_type start = 0;
        string::size_type end = 0;
137
138
        while ((end = str.find(" ", start)) != string::npos) {
139
            tokens.push_back(str.substr(start, end - start));
140
141
142
            start = end + 1;
143
        }
144
145
        tokens.push_back(str.substr(start));
146
147
        return tokens;
148 }
149
```

Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy |