Store the Water

locked

by rithvik_kolla

Editorial by rithvik_kolla

Start from the leftmost and the rightmost ends. If the leftmost end is shorter than the rightmost, imagine pouring water from the leftmost end. It fills up everything in between and does not overflow because the rightmost is always greater than the leftmost in this case (and vice-versa). In this case the rightmost end acts as a wall of a container.

Statistics

Difficulty: Medium
Time Complexity: O(n)
Required Knowledge: Basic Logic
Publish Date: Jun 26 2019

Set by rithvik_kolla

Problem Setter's code :

```
#include <stdio.h>
#include <stdlib.h>

int store(const int*, int);

int main()
{
    int t, n, sum = 0;
    scanf("%d", &t);
    while(t--)
    {
        scanf("%d", &n);
        int arr[n];
        for(int i = 0; i < n; i++)
        {
            scanf("%d", &arr[i]);
        }
        sum = store(arr, n);
        printf("%d\n", sum);
    }
}

int store(const int* arr, int n)
{
    int left = 0;
    int right = n - 1;
    int res = 0;
    int maxleft = 0, maxright = 0;
    while(left <= right)
    {
        if(arr[left] <= arr[right])
        {
            if(arr[left] >= maxleft)
                maxleft = arr[left];
            else
                res += (maxleft - arr[left]);
            left++;
        }
        else
        {
            if(arr[right] >= maxright)
                maxright = arr[right];
            else
                res += (maxright - arr[right]);
            right--;
        }
```

```
    }
    return res;
}
```