Quiz PBO C – Soal 1

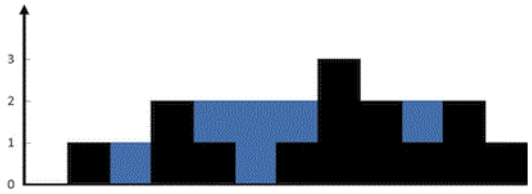
Brendan Timothy Mannuel

5025221177

Trapping Rain Water

Given n non-negative integers representing an elevation map where the width of each bar is 1.

Compute how much water it can trap after raining.



Pada soal ini kita diperlukan mencari merapa banyak air (kotak) yang tertampung diantara kolom kolom yang terbentuk dari kotak kotak. Kita akan menggunakan implementasi Stack untuk mengerjakan soal ini.

class template

std::stack <stack>

```
template <class T, class Container = deque<T> > class stack;
```

LIFO stack

Stacks are a type of container adaptor, specifically designed to operate in a LIFO context (last-in first-out), where elements are inserted and extracted only from one end of the container.

stacks are implemented as ***container adaptors***, which are classes that use an encapsulated object of a specific container class as its ***underlying container***, providing a specific set of member functions to access its elements. Elements are ***pushed***/***popped*** from the ***"back"*** of the specific container, which is known as the ***top*** of the stack.

The underlying container may be any of the standard container class templates or some other specifically designed container class. The container shall support the following operations:
`empty`
`size`
`back`
`push_back`
`pop_back`

The standard container classes `vector`, `deque` and `list` fulfill these requirements. By default, if no container class is specified for a particular `stack` class instantiation, the standard container `deque` is used.

Template parameters

`T`
Type of the elements.
Aliased as member type `stack::value_type`.

`Container`
Type of the internal ***underlying container*** object where the elements are stored.
Its `value_type` shall be `T`.
Aliased as member type `stack::container_type`.

Setelah kita coba membaca soal lagi, kita dapat melihat agar air dapat terperangkap maka kita memerlukan 2 kolom yang terdapat minimal jarak 1 diantara 2 kolom tersebut, serta ketinggian dari air ditentukan oleh kolom yang lebih rendah. Kita akan menggunakan sample testcase 2 untuk menggambarkannya. Kita akan menyimpannya di suatu array.

```
6
4 2 0 3 2 5
```
Untuk step pertama kita akan membuat suatu stack kosong. Kemudian kita akan memasukan index nya sebagai penunjuk ke tinggi agar nanti kita dapat lebih mudah mencari jarak diantara. Angka pertama akan langsung dimasukan karena stack masih kosong.

0					
4					

Kemudian untuk angka selanjutnya kita akan mengecek dengan parameter apakah stack tidak kosong dan ketinggian dari kolom di stack.top() itu lebih pendek dari ketinggian dari isi array selanjutnya. Ternyata masih belum mememnuhi sehnigga kita akan mengepush lagi.

0	1				
4	2				

Kemudian kita akan melakukan hal yang sama lagi dengan mengecek parameter, ternyata masih salah sehingga kita akan mengepush lagi.

0	1	2			
4	2	0			

Setelah itu kita akan mengechek lagi, dan ternyata ketinngian selanjutnya lebih tinggi maka kita akan menyimpan stack.top() sekarang dan kita akan mengepop nya.

Tingiatas = 0

Index skrg = 3

0	1				
4	2				

Kemudian kita akan melihat apakah stack telah kosong, ternyata tidak. Selanjutnya kita akan mencari jarak diantara kedua kolom dengan mengurangi index nya

jarakantara = i - st.top() – 1

3 – 1 – 1 = 1 (jarak antara)

Kemudian kita akan mencari kolom yang lebih rendah diantara st.top() dan ketinggian dari index sekarang. Ternyata lebih rendah st.top() karena memiliki tinggi 2 sedangkan index sekarang memiliki ketinggian 3. Kemudian kita akan melakukan pengurangan dengan angka yang telah kita simpan (mengantisipasi bila didalam kolom tersebut ada balok yang dibawahnya) yaitu 0, 2 – 0 = 0. Kemudian kita akan mengalikan nya dengan jarak yaitu 2 x 1 = 2. Kemudian kita akan menyimpan hasil ini dan melanjutkan iterasi.

Ans = 2.

Kemudian kita tetap akan mengepush index ini.

0	1	3			
4	2	3			

Kita akan melakukan pengecekan untuk index selanjutnya. Ternyata lebih rendah dari stack.top() maka kita akan mengepush nya.

0	1	3	4		
4	2	3	2		

Selanjutnya kita akan melakukan pengecekan lagi, ternyata tingginya lebih tinggi dari stack.top() yaitu 5. kita akan menyimpan stack.top() sekarang dan kita akan mengepop nya.

Tingiatas = 2

Index skrg = 5

0	1	3			
4	2	3			

Kemudian kita akan melihat apakah stack telah kosong, ternyata tidak. Selanjutnya kita akan mencari jarak diantara kedua kolom dengan mengurangi index nya

jarakantara = i - st.top() – 1

5 – 3 – 1 = 1 (jarak antara)

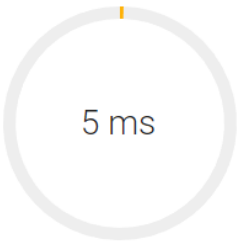
Kemudian kita akan mencari kolom yang lebih rendah diantara st.top() dan ketinggian dari index sekarang. Ternyata lebih rendah st.top() karena memiliki tinggi 3 sedangkan index sekarang memiliki ketinggian 5. Kemudian kita akan melakukan pengurangan dengan angka yang telah kita simpan (mengantisipasi bila didalam kolom tersebut ada balok yang dibawahnya) yaitu 0, 3 – 2 = 1. Kemudian kita akan mengalikan nya dengan jarak yaitu 1 x 1 = 1. Kemudian kita akan menyimpan hasil ini dan melanjutkan iterasi.

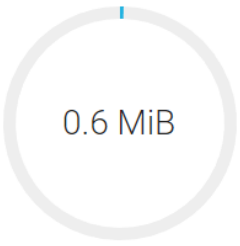
Ans = 3.

Ternyata stack belum kosong sehingga kita akan melakukan pengecekan terus hingga stcak kosong kemudian kita tingal mengoutput ans

```cpp
#include <stdio.h>
#include <stack>
using namespace std;

int main()
{
    int n;
    stack <int> st;
    scanf("%d", &n);
    int array[n];
    for(int i = 0; i < n; i++){
        scanf("%d", &array[i]);
    }
    int m = sizeof(array) / sizeof(array[0]);

    int ans = 0;

    for(int i = 0; i < m; i++){
        while(!st.empty() && (array[st.top()] < array[i])){

            int tinggiatas = array[st.top()];
            st.pop();

            if (st.empty()) break;

            int jarakantara = i - st.top() - 1;

            int tinggi = min(array[st.top()], array[i]) - tinggiatas;

            ans += tinggi * jarakantara;
        }
        st.push(i);
    }

    printf("%d", ans);

    return 0;
}

```

Problem	Submitted	Programming Language	Author
Trapping Rain Water	1 hour ago	C++ 20 (gnu 10.2)	Brendan_C177





Your submission was graded C, which means it passed all tests and used LESS resources than 50% of the submissions on the website.