

Nama : Bryan Yehuda Mannuel
NRP : 05311940000021

Latihan Modul 4 Sesi Lab

Mata Kuliah Struktur Data

1. Deque (<https://www.hackerrank.com/challenges/deque-stl/problem>)

Source Code

```
#include<bits/stdc++.h>
#include <deque>
using namespace std;

//By : Bryan Yehuda Mannuel
//NRP : 05311940000021

void KMax(int arr[], int x, int y)
{
    deque<int> De_que; //pemanggilan fungsi deque
    int i;

    for(i=0; i<y; i++) //lakukan perulangan untuk mengikat subarray
    {
        while(!De_que.empty() && arr[i]>=arr[De_que.back()]) //jika tidak kosong, deque dan index array yg akan
        { //dimasukan lebih besar dari yang ada di belakang
            De_que.pop_back(); //hapus index belakang
        }
        De_que.push_back(i); //secara default jika kosong akan memasukan posisinya dibelakang
        //agar dapat dibandingkan dengan index array yang baru masuk ke deque
    }

    for(i=y; i<x; i++) //perulangan lanjutan untuk mendapatkan nilai tertinggi dari range subarray
    {
        cout<<arr[De_que.front()]<<" "; //print yang ada di depan karena paling besar
        while(!De_que.empty() && De_que.front()<=i-y) //apabila deque, isi dan nilai front deque itu kurang dari i-y
        {
            De_que.pop_front(); //akan menghapus posisi paling depan
        }
        while(!De_que.empty() && arr[i]>=arr[De_que.back()]) //jika deque isi dan nilai array index lanjutan itu
        //lebih besar dari yang di back
        {
            De_que.pop_back(); //maka hapus yang di back
        }
        De_que.push_back(i); //secara default index array lanjutan akan ditaruh di belakang deque
    }
    cout<<arr[De_que.front()]<<endl; //tampilkan hasil semua yang terbesar dari tiap subarray
}

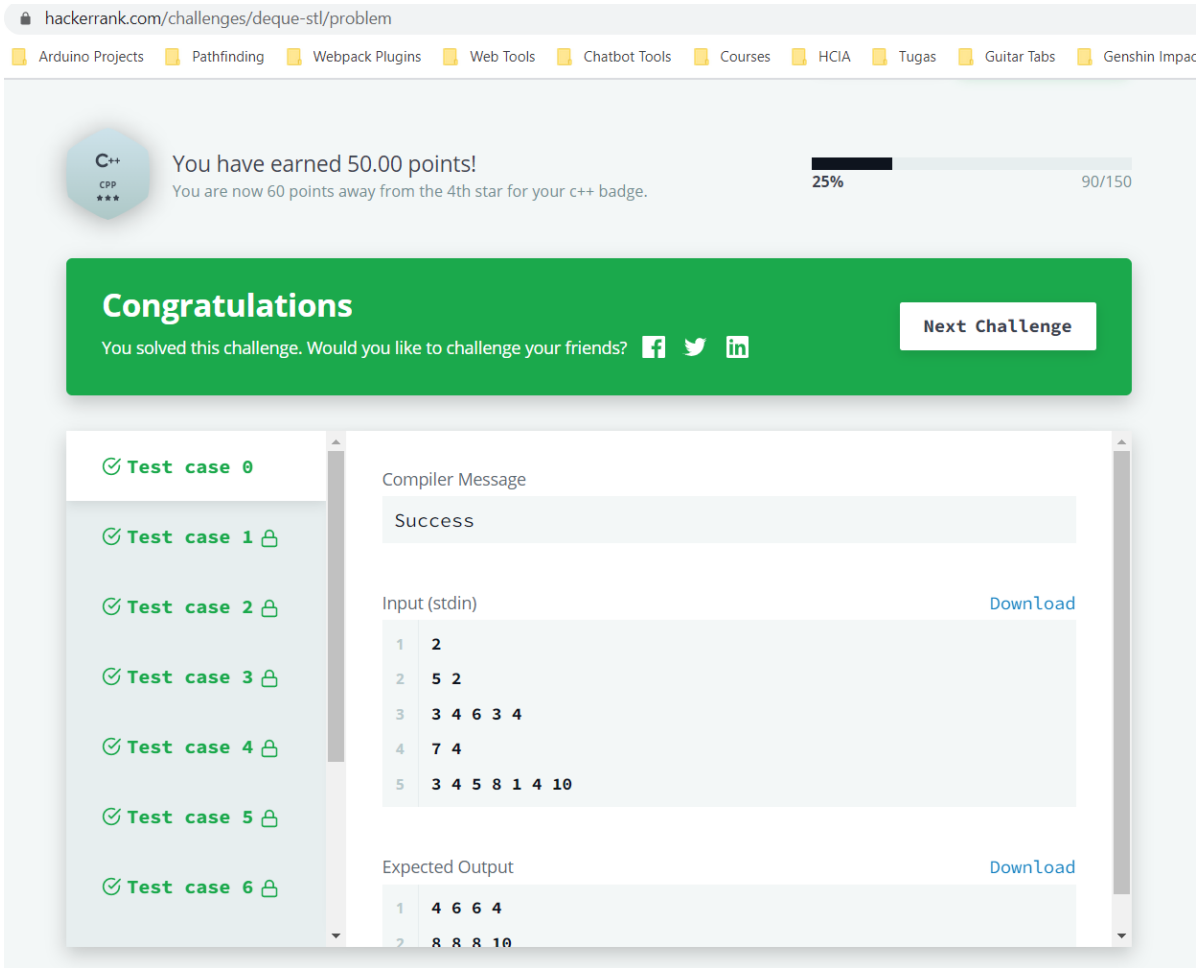
int main()
{
    int t;
    scanf("%d",&t);
    while(t--)
    {
        int x,y;//inisialisasi variable untuk menerima banyaknya array dan size subarraynya
        cin>>x>>y;
        int i;
        int * arr=new int [x]; //inisialisasi pointer array
        for(i=0;i<x;i++) //perulangan inputan user hingga memenuhi batas array yang telah didefinisikan
            cin>>arr[i];
        KMax(arr, x, y); //parsing array data, banyaknya range array, dan size sub arraynya.
    }
    return 0;
}
```

Test Run

```
2
5 2
3 4 6 3 4
4 6 6 4
7 4
3 4 5 8 1 4 10
8 8 8 10

-----
Process exited after 48.36 seconds with return value 0
Press any key to continue . . .
```

Bukti AC



Analisis

Deque merupakan sebuah jenis queue yang memiliki dua ujung yaitu, depan dan belakang. Algoritma yang digunakan untuk menyelesaikan soal ini yaitu, pertama-tama pengguna menginputkan berapa banyak test case yang diinginkan, kedua pengguna memasukan berapa banyak kapasitas index array yang dapat ditampung, dan ketiga program akan memecahnya dalam ukuran subarray sesuai dengan input user.

Ketika ketiga hal tersebut telah kita lakukan maka selanjutnya adalah saatnya untuk mengelompokkan array (sesuai dengan kapasitas array) menjadi sekelompok yang lebih kecil dalam sub-subarray, lalu dari masing-masing subarray yang ada, kita akan bandingkan mana yang terbesar yang kemudian akan ditaruh keposisi front deque dan di print. Dalam pembagian sub-subarray, dibuat sebanyak mungkin array yang dapat dibagi. Tentunya hal-hal tersebut disesuaikan dengan batas maksimal yang telah ditentukan pada *constraint* yang telah diinputkan oleh user.

Sebagai contoh untuk memodelkan dari persoalan diatas adalah:

1
7 4
3 4 5 8 1 4 10

1 disini merupakan banyaknya test case yang akan diuji dan 7 merupakan size of array yang dapat ditampung. 4 merupakan size of sub-array yang nanti akan kita pecah. Hasil pemecahan akan menjadi seperti ini {3,4,5,8},{4,5,8,1}, {5,8,1,4}, dan {8,1,4,10}. Sehingga setelah dilakukan algoritma pencarian nilai terbesar dari tiap-tiap sub-array akan menghasilkan hasil sebagai berikut 8 8 8 10 .

Nama : Bryan Yehuda Mannuel
NRP : 05311940000021

2. Vector Erase (<https://www.hackerrank.com/challenges/vector-erase/problem>)

Source Code

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;

//By : Bryan Yehuda Mannuel
//NRP : 05311940000021

int main() {
    int x,y,a,b,c,size;
    cin>>x;
    vector<int> z; //define vector
    for(y=0;y<x;y++)
    {
        int x;
        cin>>x;
        z.push_back(x);
    }
    cin>>a>>b>>c; //inputan a sebagai nilai mana yang akan dihapus index ke berapa
                //inputan b untuk sebagai penentu awal range yang kaan dihapus
                //inputan c untuk sebagai penentu akhir dari range yang akan dihapus
    z.erase(z.begin()+a-1); //eksekusi hapus pada suatu index tertentu
    z.erase(z.begin()+b-1,z.begin()+c-1); //eksekusi hapus pada suatu range

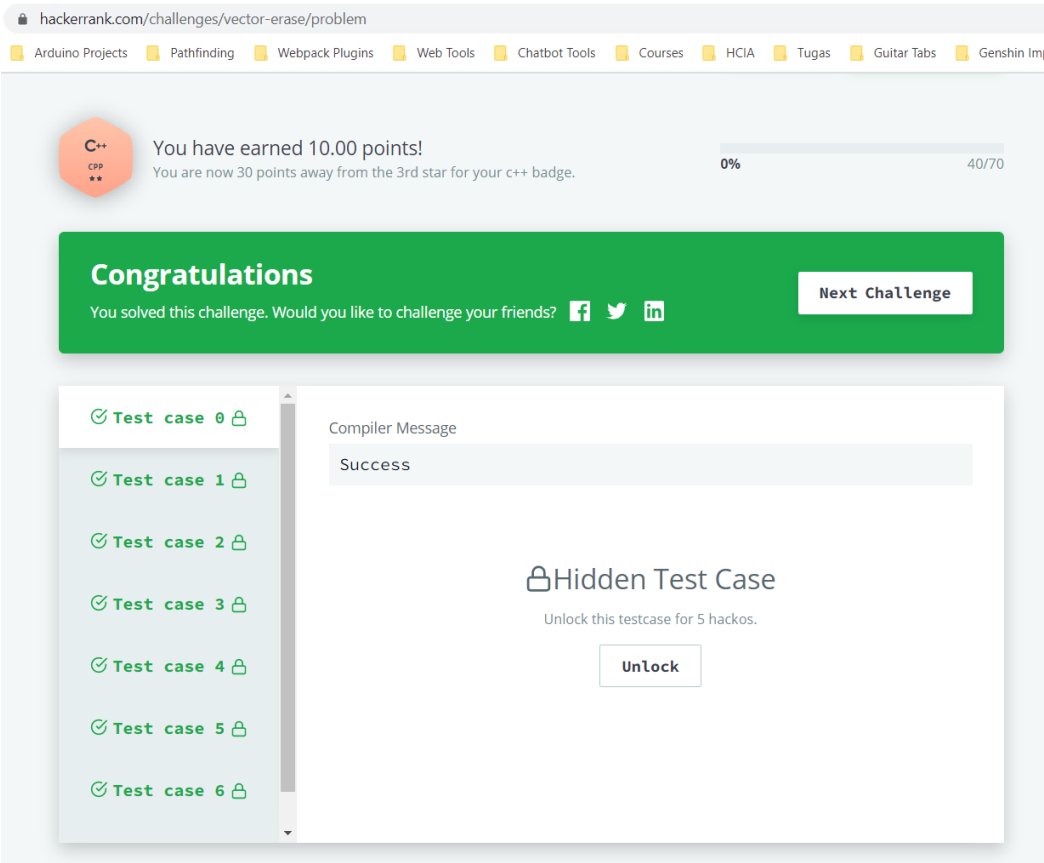
    size=z.size(); //menata ulang banyaknya angka yang tersisa

    cout<<size<<endl; //print jumlah total ada berapa angka saat ini
    for(y=0;y<size;y++) //perulangan untuk menghasilkan daftar angka yang tersedia
        cout<<z[y]<<" ";
    return 0;
}
```

Test Run

```
6
1 4 2 6 8 9
2
2 4
3
1 8 9
-----
Process exited after 55.75 seconds with return value 0
Press any key to continue . . .
```

Bukti AC



Nama : Bryan Yehuda Mannuel
NRP : 05311940000021

Analisis

Soal ini bertema tentang Vector pada C++ khususnya tentang penghapusan. Hal ini jelas terlihat ketika soal mengarahkan kita untuk menggunakan vector erase baik itu terlihat dari index yang dituju ataupun range dari suatu vector yang akan dihapus. Cara kerja dari program ini adalah, pertama-tama kita meminta inputan kepada user tentang berapa banyak angka integer yang akan dimasukkan, kedua kita meminta kepada user detail dari integer dengan batas yang telah ditentukan sebelumnya, ketiga kita akan tentukan pada indeks beberapa integer pada vector akan dihapus dan juga pada range penghapusannya (dari indeks ke A hingga ke B). Ketika semua sudah dilakukan, maka program akan menampilkan hasilnya.

3. Sets (<https://www.hackerrank.com/challenges/cpp-sets/problem>)

Source Code

```
#include <iostream>
#include <set>
using namespace std;

//BY : Bryan Yehuda Mannuel
//NRP : 05311940000021

int main()
{
    set<int> s; //deklarasi set dengan nama s value int
    int q;
    cin >> q;

    while (q > 0)
    {
        int y, x;
        cin >> y >> x; //masukan perintah tambahkan, hapus, atau display dan target nilai angkanya

        if (y == 1) //memfilter tiap query
        {
            s.insert(x); //inputkan angka
        }
        else if (y == 2)
        {
            s.erase(x); //hapus angka
        }
        else if (y == 3)
        {
            if (s.find(x) == s.end()) //mencari apakah value x ada atau tidak
            {
                cout << "No" << endl; //jika tidak ada
            }
            else
            {
                cout << "Yes" << endl; //jika ada
            }
        }
        q--;
    }
    return (0);
}
```

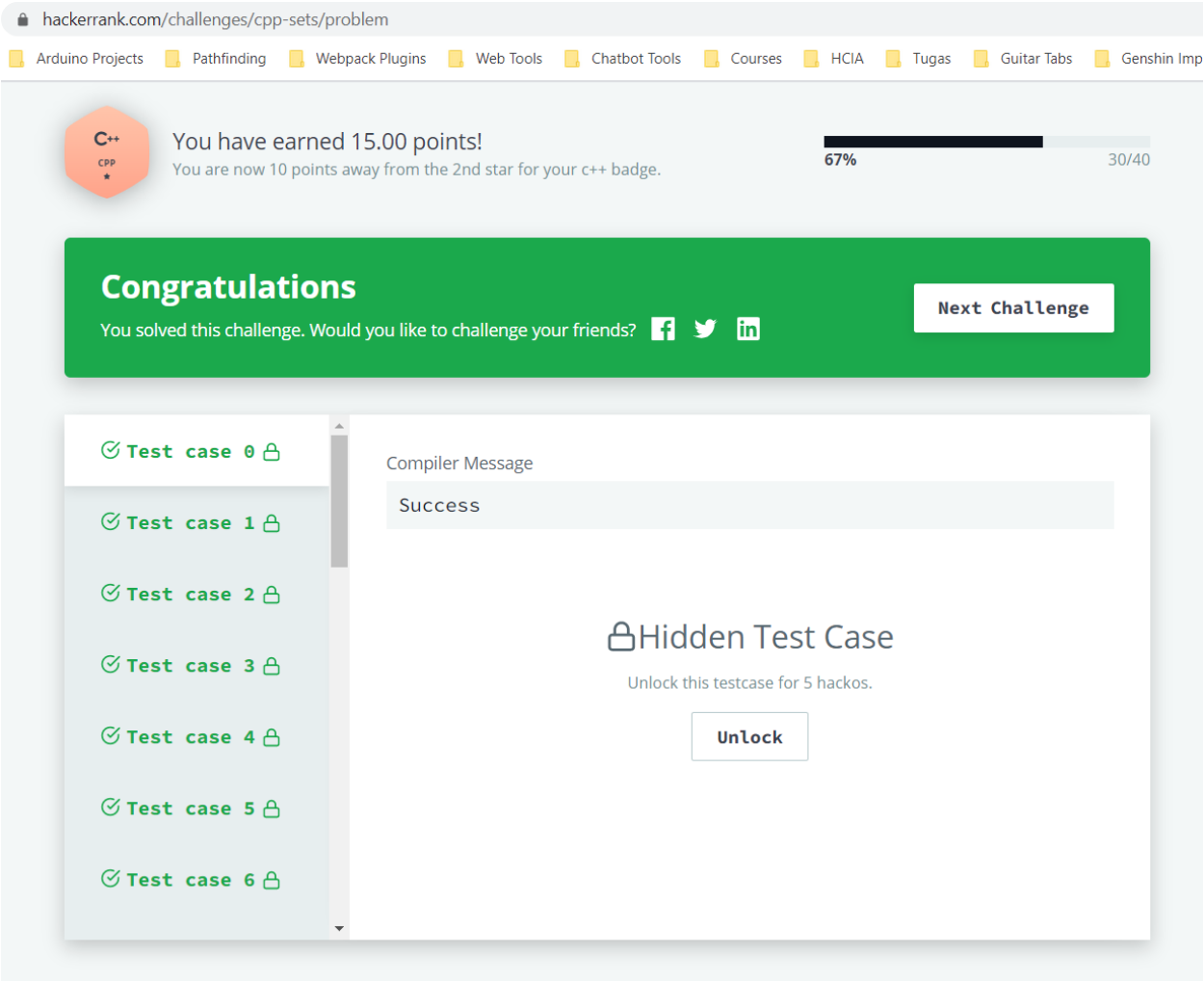
Test Run

```
8
1 9
1 6
1 10
1 4
3 6
Yes
3 14
No
2 6
3 6
No

-----
Process exited after 49.24 seconds with return value 0
Press any key to continue . . . █
```

Nama : Bryan Yehuda Mannuel
NRP : 05311940000021

Bukti AC



Analisis

Kita deklarasikan sebuah set bernama s . Kemudian, program akan menerima sebuah integer bernama q. Setelah itu program akan melakukan looping sebanyak q kali untuk menerima 2 integer, yakni y dan x. y merupakan tipe query/perintah yang akan dijalankan, lalu x merupakan integer yang akan diproses oleh querynya.

Tipe-tipe query y yakni :

- 1 : Menambahkan x kedalam set
- 2 : Menghapus x dari set
- 3 : Mencari x dalam set. Jika ditemukan, program akan print kata “Yes”, jika tidak akan print kata “No”.

4. Maps (<https://www.hackerrank.com/challenges/cpp-maps/problem>)

Source Code

```
#include <iostream>
#include <map>
using namespace std;

//BY : Bryan Yehuda
//NRP : 05311940000021

int main()
{
    map<string, int> m; //Inisialisasi map dengan tipe data string dan integer dengan nama m
    int q;
    cin >> q;

    while (q > 0)
    {
        int type;
        string x;
        cin >> type >> x;

        if (type == 1) //jika 1 maka akan memberikan tanda nilai Y terhadap siswa X
        {
            int y;
            cin >> y;
```

Nama : Bryan Yehuda Mannuel
NRP : 05311940000021

```
        if (m.find(x) == m.end()) //mencari index X terlebih dahulu
        {
            m.insert(make_pair(x, y)); //jika tidak ada maka buat index terlebih dahulu
        }
        else
        {
            int old_val; //jika ada, maka tambahkan value dari map sebelumnya
            old_val = m[x];
            m.erase(x);
            int new_val = old_val + y;
            m.insert(make_pair(x, new_val));
        }
    }
    else if (type == 2) //jika 2 maka akan menghapus nilai dari siswa X
    {
        m.erase(x);
    }
    else if (type == 3) //jika 3 maka print value
    {
        cout << m[x] << "\n";
    }
    q--;
}

return (0);
}
```

Test Run

```
7
1 Jesse 20
1 Jess 12
1 Jess 18
3 Jess
30
3 Jesse
20
2 Jess
3 Jess
0
-----
Process exited after 155.5 seconds with return value 0
Press any key to continue . . .
```

Bukti AC

hackerrank.com/challenges/cpp-maps/problem

Arduino Projects

Pathfinding

Webpack Plugins

Web Tools

Chatbot Tools

Courses

HCIA

Tugas

Guitar Tabs

Genshin Impact

Upload Code as File

Test against custom input

Run Code

Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends?

Facebook

Twitter

LinkedIn

Next Challenge

Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Compiler Message

Success

Hidden Test Case

Unlock this testcase for 5 hacks.

Unlock

Nama : Bryan Yehuda Mannuel
NRP : 05311940000021

Analisis

Kita deklarasikan sebuah map dengan key type string dan data type integer. Program akan menerima sebuah integer bernama q. Lalu program akan melakukan looping sebanyak q kali untuk menerima integer bernama type, dan string bernama x. Tiap type ini akan melakukan tugasnya masing masing yakni

- 1 – menambahkan sebuah value kedalam map index y. Karena itu akan diinputkan integer tambahan bernama y. Lalu program akan melakukan insert ke dalam map dengan y sebagai value dari map index x. Program nantinya bisa saja harus menuliskan query tipe ini kedalam index yang sama. Karena itu akan dicek terlebih dahulu apakah index yang ingin ditambahkan valuenya sudah ada atau belum. Jika belum, maka akan dimasukkan biasa. Jika sudah, maka value dari index x yang lama (old_val akan ditambahkan dengan value yang baru (y) menjadi integer bernama new_val, lalu dihapus terlebih dahulu map yang lama, lalu di insert kan new_val ke map index x
- 2 – menghapus value dari index x
- 3 – mencetak value dari index x