1. Two Sum II - Input array is sorted

Given a sorted array of integers, return the indices of the two numbers such that they add up to a specific target.

Code:  
/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

  GDB online is an online compiler and debugger tool for C, C++, Python, PHP, Ruby,

  C#, OCaml, VB, Perl, Swift, Prolog, Javascript, Pascal, COBOL, HTML, CSS, JS

  Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <bits/stdc++.h>

using namespace std;

vector<int> twosumres(int t, vector<int> arr){

    int i = 0;

    int j = arr.size()-1;

    while(i<j){

        int x = arr[i]+arr[j];

        if(x==t){

            return {i,j};

        }

        if(x>t){

            j--;

        }

        else{

            i++;

        }

    }

    return {-1};

}

int main()

{

    vector<int> arr{0,1,2,3,3,7};

    vector<int> res = twosumres(6, arr);

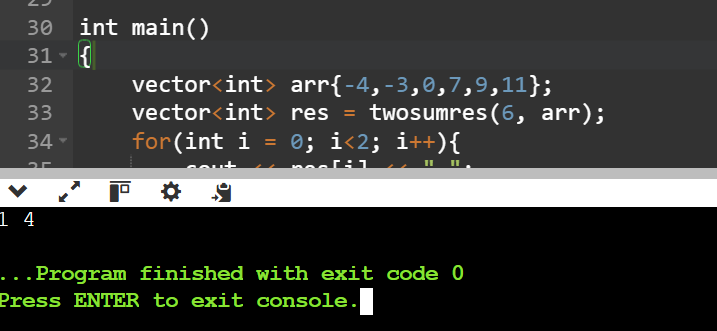
    for(int i = 0; i<2; i++){

        cout << res[i] << " ";

    }

    return 0;

}



2. Subarray Sum Equals K

Given an array of integers and a target sum k, return the total number of continuous subarrays whose sum equals to k.

Code:

#include <bits/stdc++.h>

using namespace std;

int findsuba(vector<int> arr, int tar){

    int count = 0;

    unordered\_map<int, int> freq;

    freq[0] = 1;

    int sum = 0;

    for(int i = 0; i<arr.size(); i++){

        sum = sum + arr[i];

        if(freq.find(sum - tar)!=freq.end()){

            count+= freq[sum-tar];

        }

        if(freq.find(sum)==freq.end()){

            freq[sum] = 1;

        }

        else{

            freq[sum]++;

        }

    }

    return count;

}

int main()

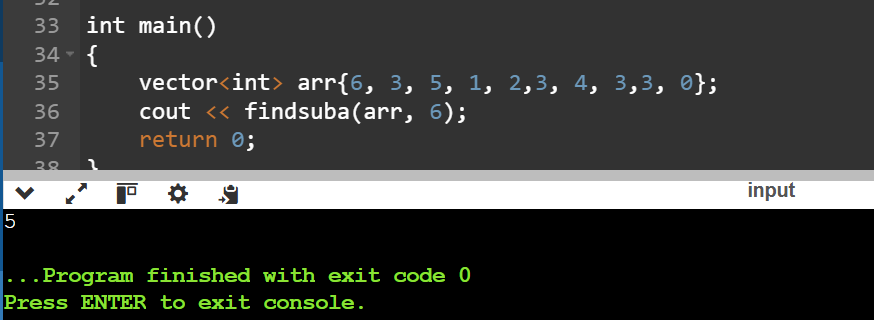
{

    vector<int> arr{6, 3, 5, 1, 2,3, 4, 3,3, 0};

    cout << findsuba(arr, 6);

    return 0;

}



3. Next Greater Element

Given a circular array, find the next greater number for every element.

Code:

#include <bits/stdc++.h>

using namespace std;

vector<int> nexag(vector<int> arr){

    next\_permutation(arr.begin(), arr.end());

    return arr;

}

int main()

{

    vector<int> arr{2,3,4};

    arr = nexag(arr);

    for(int i = 0; i<arr.size(); i++){

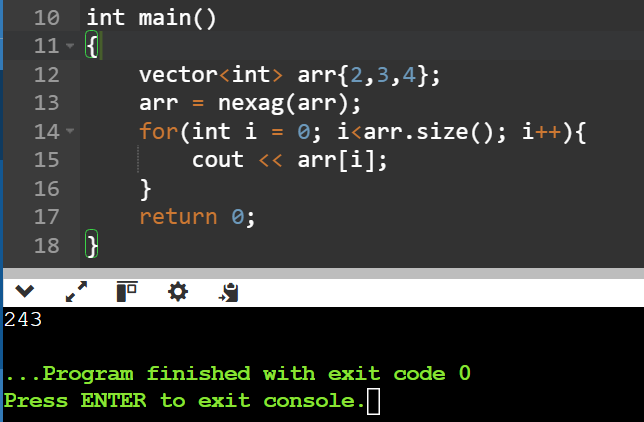
        cout << arr[i];

    }

    return 0;

}

Output:



7. Strings

Longest Substring Without Repeating Characters

Given a string, find the length of the longest substring without repeating characters.

#include <bits/stdc++.h>

using namespace std;

string longestwithoutrep(string x){

    int n = x.length();

    unordered\_set<char> st;

    string z  = "";

    string mins = "";

    for(int i = 0; i<n; i++){

        if(st.find(x[i])==st.end()){

            st.insert(x[i]);

            z = z + x[i];

        }

        else{

            st = {};

            if(z.length()>mins.length()){

                mins = z;

            }

            z = "";

        }

    }

    if(z.length()>mins.length()){

                mins = z;

    }

    return mins;

}

int main()

{

    cout << longestwithoutrep("abcdaghfikg");

}

A screenshot of a computer

AI-generated content may be incorrect.