Om Thanage - 16010123217

Fibonacci Series:

#include <bits/stdc++.h>

using namespace std;

#define int long long

#define endl "\n"

const int MOD = 1e9 + 7;

const int INF = LLONG\_MAX >> 1;

void fibo(int n){

    int a = 0, b = 1;

    cout<<a<<" "<<b<<" ";

    for(int i = 2; i < n; i++){

        int c = a + b;

        cout<<c<<" ";

        a = b;

        b = c;

    }

    cout<<endl;

}

signed main(){

    ios::sync\_with\_stdio(false); cin.tie(NULL);

    int n;  cin>>n;

    if(n < 0){

        n -=(2\*n);

    }

    fibo(n);

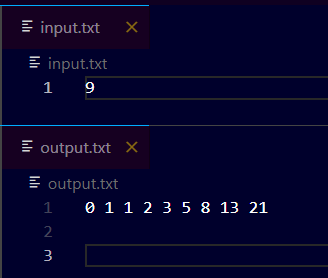
    cout<<endl;

    return 0;

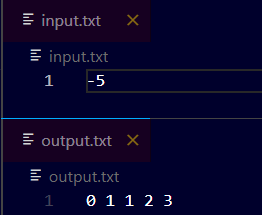
}

Output:

Testcase 1:



Testcase 2:



Palindrome

#include <bits/stdc++.h>

using namespace std;

#define int long long

#define endl "\n"

const int MOD = 1e9 + 7;

const int INF = LLONG\_MAX >> 1;

bool palindrome(string s){

        int n = s.size();

        int l = 0, r = n-1;

        while(l < r){

            if(s[l] != s[r]){

                return false;

            }

            else{

                l++;

                r--;

            }

        }

    return true;

}

signed main(){

    ios::sync\_with\_stdio(false); cin.tie(NULL);

        int n; cin>>n;

        vector<string> s;

        for(int i = 0; i < n; i++){

            string temp;

            cin>>temp;

            s.push\_back(temp);

        }

        for(auto i: s){

            if(palindrome(i)){

                cout<<"Yes"<<endl;

            }

            else{

                cout<<"No"<<endl;

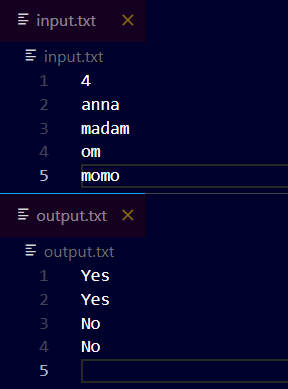
            }

        }

    return 0;

}

Output:



Array Rotation:

#include <bits/stdc++.h>

using namespace std;

#define int long long

#define endl "\n"

const int MOD = 1e9 + 7;

const int INF = LLONG\_MAX >> 1;

void arrRotation(vector<int> &a, int pos, char dir){

    int n  = a.size();

    switch(dir){

        case 'a':

            for(int i = 0; i < pos; i++){

                int temp = a[0];

                for(int j = 0; j < n-1; j++){

                    a[j] = a[j+1];

                }

                a[n-1] = temp;

            }

            break;

        case 'c':

            for(int i = 0; i < pos; i++){

                int temp = a[n-1];

                for(int j = n-1; j > 0; j--){

                    a[j] = a[j-1];

                }

                a[0] = temp;

            }

            break;

        default:

            cout<<"Invalid direction"<<endl;

            return;

    }

    for(auto i: a){

        cout<<i<<" ";

    }

}

signed main(){

    ios::sync\_with\_stdio(false); cin.tie(NULL);

    int n; cin>>n;

    vector<int> a(n);

    for(int i = 0; i < n; i++)  cin>>a[i];

    int pos;    cin>>pos;

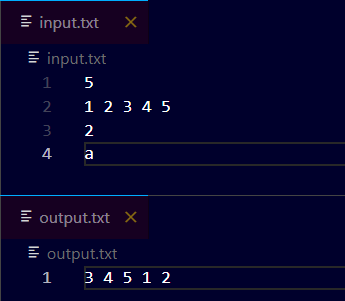
    char dir;   cin>>dir;

    arrRotation(a, pos, dir);

    return 0;

}

Output: Anticlockwise



Clockwise:

