// header file.......................

#include <iostream.h> // cin, cout

#include <conio.h> // getch(), clrscr()

#include <stdio.h> // gets(), strlen()

// #include <string.h> // strlen() karena tidak berguna tidak saya pakek, udah menggunakan <stdio.h>

#define maxkal 200 // definisi konstanta maxkal (maksimal kalimat) dengan nilai 200

// deklarasi struct..................

struct tumpukan // tipe struct dengan nama tumpukan

{

    char isi[maxkal]; // data didalam struct tumpukan yang memiliki tipe data char dengan nama isi jumlah char/huruf maxkal = 200

    int atas; // data didalam struct tumpukan yang memiliki tipe data int dengan nama atas

};

tumpukan tumkal; // struct tumpukan didefiniskan lagi dengan nama tumkal (tumpukan kalimat)

// fungsi push................................

// fungsi untuk mendorong/menekan data.

void push(char x) {

    if(tumkal.atas == maxkal) { // jika kondisi tumkal.atas sama dengan maxkal jalankan block kode dibawah ini

        cout << "\nTumpukan Sudah Penuh !"; // tampilkan ke konsol/terminal

        getch(); // menghentikan sementara sampai salah satu tombol keyboard ditekan.

    } else { // jika kondisi diatas salah maka jalankan block kode dibawah ini

        tumkal.atas = tumkal.atas + 1; // top = top + 1.

        tumkal.isi[tumkal.atas] = x; // isi[top] = x

    }

}

// fungsi pop................................

// fungsi untuk mengambil/meletuskan/mengeluarkan data.

char pop() {

    char hasil; // varibel hasil bertipe data char

    if(tumkal.atas == 0) { // jika kondisi tumkal.atas sama dengan 0 (nol) maka jalankan block kode dibawah ini

        cout << "\nTumpukan sudah kosong !"; // tampilkan ke konsol/terminal

        hasil = ' '; // nilai hasil sama dengan ' '.

    } else { // jika kondisi diatas tidak terpenuhi maka jalankan block kode dibawah ini

        hasil = tumkal.isi[tumkal.atas]; // nilai hasil = isi[top].

        tumkal.atas = tumkal.atas - 1; // top = top - 1.

    }

    return hasil; // kembalikan nilai hasil dari percabangan kondisi diatas.

}

// program utama.............................

void main() {

    int i; // variabel global i bertipedata int

    char kalimat[maxkal]; // variabel global kalimat bertipedata char dengan jumlah huruf (index) = 200.

    clrscr(); // clear screen (bersihkan layar) pada konsol/terminal

    tumkal.atas = 0; // top = 0.

    cout << "Aplikasi stack untuk membalik kalimat\n"; // tampilkan ke konsol/terminal

    cout << "-----------------------------------------\n\n"; // tampilkan ke konsol/terimanl

    // Inputkan kalimat yang akan dibalik.........................

    cout << "Masukkan sembarang kalimat : "; // tampilkan ke konsol/terminal

    gets(kalimat); // masukan data "kalimat", menggunakan gets supaya bisa menggunakan spasi.

    clrscr(); // clear screen (bersihkan layar) pada konsol/terminal

    cout<<"\nKalimat Awal : " << kalimat; // tampilkan ke konsol terminal data "kalimat" yang dimasukan tadi.

    // proses awal balik kalimat.................................

    for(i = 0; i < strlen(kalimat); i++) { // perulangan untuk memasukan data ke dalam tumpukan, menggunakan strlen (string length/panjang kata) untuk mengetahui panjang kata atau kalimat yang dimasukan tadi, diulangi sesuai panjang kalimat/kata yang dimasukan.

        push(kalimat[i]); // memasukan data ke tumpukan.

    }

    cout << "\n\nKalimat setelah di balik : "; // tampilkan ke konsol/terminal

    for(i = 0; i < strlen(kalimat); i++) { // perulangan untuk menampilkan data "kalimat", diulangi sesuai panjang kalimat/kata yang dimasukan.

        cout << pop(); // menampilkan data "kalimat" yang sudah dibalik.

    }

    getch(); // menghentikan sementara sampai salah satu tombol keyboard ditekan.

}