

Stack

Tujuan Praktikum

1. Memahami Stack dalam program,
2. Mampu menerapkan Stack untuk menyelesaikan berbagai kasus.

Dasar Teori

“A stack is an ordered collection of items into which new items may be inserted and from which items may be deleted at one end, called top of the stack” * Yedidyah L, Moshe J. A., and Aaron M.

Tenenbaum; Data Structures Using C and C++. Secara sederhana, tumpukan bisa diartikan sebagai suatu kumpulan data yang seolah-olah ada data yang diletakan diatas data yang lain. Satu hal yang perlu kita ingat adalah bahwa kita bisa menambah (menyisipkan) data, dan mengambil (menghapus) data lewat ujung yang sama, yang disebut sebagai ujung atas tumpukan (*top of stack*).

LIFO (Last In First Out) adalah sifat dari stack data yang disimpan terakhir akan diambil lebih dahulu, data yang disimpan pertama kali akan diambil paling akhir”

Praktik (Guided)

Kompilasi program berikut ini dan amati outputnya pada layar Anda. Perhatikan baik-baik pemanggilan dan penggunaan fungsi-fungsi serta prosedurnya agar dapat mengerjakan tugas yang diberikan!

```
#include <iostream>
#include <stdlib.h> // menggunakan system("cls")
#include <conio.h> // menggunakan getch()
#define MAX 10 // batas stack yang ingin dibuat, boleh lebih
using namespace std;

int top=-1, Stack[MAX];

void push(){
    if(top == MAX-1){
        cout << ">> Stack sudah Penuh !" << endl;
    }else{
        top++;
        cout << ">> Masukan Data : ";
        cin >> Stack[top];
        cout << "\n Pada Index ke " << top << "\n";
        cout << "Data [" << Stack[top] << "] Telah Ditambah !" << endl;
    }
}

void pop(){
    if(top == -1){
        cout << ">> Stack Kosong !" << endl;
    }else{

```

```

        cout << "\n Data [" << Stack[top] << "]" pada index ke '" << top << "' dalam Stack Diambil !" << endl;
        Stack[top--];
    }
}

int clearr(){
    return top = -1;
}

void Print(){
    if (top == -1) {
        cout << "          Stack : ";
        cout << "\n===== " << endl;
        cout << "          Empty ! \n===== " << endl;
    }
    else {
        cout << "          Stack : ";
        cout << "\n===== " << endl;
        for (int i = top; i >= 0; i--){
            cout << "      XXXXX[ " << Stack[i] << " ]XXXXX" << endl;
        }
        cout << "===== " << endl;
    }
}

int main()
{
    int choose;
    do {
        Print();
        cout << "\n1. Push"
            << "\n2. Pop"
            << "\n3. Clear"
            << "\n4. Exit"
            << "\nPergerakan ( TOP ) : " << top
            << "\n\n>> Choose : "; cin >> choose;
        switch (choose){
            case 1:
                push();getch();
                break;
            case 2:
                pop();getch();
                break;
            case 3:
                clearr();
                cout << "\nClear Success" << endl;
                getch();
                break;
            case 4:
                cout << "Terima Kasih telah menggunakan program ini :)" << endl;
                getch();
                break;
            default:
                cout << "Pilihan tidak ada\nMasukan pilihan sesuai dengan daftar" << endl;
                getch();
                break;
        }
        system("cls");
    }while(choose !=4);
}

```

Latihan (Unguided)

1. Buatlah program untuk melakukan pembalikan terhadap kalimat dengan menggunakan stack.

Contoh:

Kalimat : Struktur Data

Hasil setelah dibalik : ataD rutkurtS

2. Dari soal no 1, buatlah program untuk menentukan apakah sebuah kalimat yang diinputkan dalam program (dengan menggunakan stack) adalah sebuah palindrom atau bukan. Palindrom adalah kalimat yang jika dibaca dari depan dan dari belakang, maka bunyinya sama.

Contoh:

Kalimat : sugus

Kalimat tersebut adalah palindrom

Kalimat : tenia

Kalimat tersebut bukan palindrom

Algoritma :

- a. Mulai
 - b. Masukkan kata
 - c. Hitung jumlah hurufnya
 - d. Masukkan ke dalam stack (push)
 - e. Bandingkan elemen 1 dalam stack dengan elemen terakhir (pop)
 - f. Perbandingan dilakukan berulang sebanyak jumlah huruf
 - g. Jika huruf yang dibandingkan semuanya sama, maka kata tersebut adalahpalindrome
 - h. Selesai
3. Buatlah program dengan stack untuk mengubah notasi matematika infix menjadi postfix!