

PERTEMUAN XII

STACK (Lanjut)

TUJUAN PRAKTIKUM

- Mahasiswa dapat menjelaskan pengertian dan pembuatan Stack dengan C++.
- Mahasiswa dapat melakukan operasi penyisipan dan penghapusan elemen dalam Stack dengan C++.
- Mahasiswa dapat mengimplementasikan Stack dengan C++.

TEORI DASAR

a) Pendahuluan

Stack atau tumpukan adalah kumpulan elemen yang hanya dapat di tambah atau dihapus dari satu ujung (gerbang) yang sama. Hal ini menunjukkan bahwa seolah-olah suatu elemen diletakan di atas elemen yang lain. Yang memberi gambaran bahwa Stack mempunyai sifat LIFO (*Last In First Out*) yang berarti bahwa elemen yang terakhir masuk akan pertama keluar.

Representasi Stack dapat dilakukan menggunakan Array atau Linked List. Kedua representasi mempunyai keunggulan dan kelemahan. Dengan Array, stack juga dapat disajikan dengan Single Stack dan Double Stack.

TUGAS PRAKTIKUM

- Buatlah program Stack dengan menggunakan Singly Linked List (simpan dengan nama `lat12_1.cpp`)

```
#include<iostream.h>
#include<conio.h>
#include<stdlib.h>
#define true 1
#define false 0
typedef struct node *simpul;
struct node
{
    char Isi;
    simpul next ;
};
```

```

//=====
//==Prototype Function==
//=====
void Sisip_Belakang (simpul &L, char elemen);
void Hapus_Belakang (simpul &L);
void Cetak (simpul L);
//=====
//==Function Main==
//=====
main ( )
{
    char huruf ;
    simpul L = NULL; //Pastikan bahwa L kosong
    cout<<"Operasi Single Linked List Pada Stack==\n\n";
    //=====
    //==Sisip Belakang==
    //=====
    cout<<endl<<endl<<"Penyiapan Stack "<<endl<<endl;
    cout<<"Masukan Elemen : "; cin>> huruf;
    Sisip_Belakang (L, huruf);
    cout<<"Masukan Elemen : "; cin>> huruf;
    Sisip_Belakang (L, huruf);
    cout<<"Masukan Elemen : "; cin>> huruf;
    Sisip_Belakang (L, huruf);
    cout<<"Masukan Elemen : "; cin>> huruf;
    Sisip_Belakang (L, huruf);
    cout<<"Masukan Elemen : "; cin>> huruf;
    Sisip_Belakang (L, huruf);
    Cetak (L);
    //=====
    //==Hapus Simpul Belakang==
    //=====
    cout<<endl<<endl<<"Hapus Elemen "<<endl;
    Hapus_Belakang (L);
    Cetak (L);
    cout<<endl<<endl<<"Hapus Elemen "<<endl;
    Hapus_Belakang (L);
    Cetak (L);
    cout<<endl<<endl<<"Hapus Elemen "<<endl;
    Hapus_Belakang (L);

```

```

Cetak (L);
cout<<endl<<endl<<"Hapus Elemen "<<endl;
Hapus_Belakang (L);
Cetak (L);
getch ();
}
//*****
/**FUNCTION SISIP SIMPUL DI BELAKANG*****
//*****
void Sisip_Belakang (simpul & L, char elemen)
{
    simpul bantu, baru;
    baru= (simpul) malloc(sizeof(simpul));
    baru->isi = elemen;
    baru->next = NULL;
    if(L == NULL)
        L=baru;
    else
    {
        bantu=L;
        while(bantu->next != NULL)
            bantu=bantu->next;
        bantu->next=baru;
    }
}
//*****
/**FUNCTION HAPUS SIMPUL BELAKANG*****
//*****
void Hapus_Belakang (simpul & L)
{
    simpul bantu, hapus;
    if(L == NULL)
        cout<<"Linked List Kosong.....";
    else
    {
        bantu=L;
        while(bantu->next->next != NULL)
            bantu=bantu->next;
        hapus = bantu ->next;
        bantu->next = NULL;
        free(hapus);
    }
}

```

```

//*****
//**FUNCTION MENCETAK ISI LINKED LIST**
//*****
void Cetak(simpul L)
{
    simpul bantu;
    if (L==NULL)
        cout<<"Linked List Kosong ..... "<<endl;
    else
    {
        bantu=L;
        cout<<endl<<"Isi Linked List : ";
        while (bantu->next != NULL)
        {
            cout<<bantu->Isi <<"-";
            bantu=bantu->next;
        }
        cout<<bantu->Isi;
    }
} //===== eof=====

```

- b) **Buatlah program menu untuk menampilkan program-program pada pertemuan XI dan XII, menggunakan perintah IF (simpan dengan nama lat12_2.cpp)**

TUGAS PENDAHULUAN

1. Jelaskan perbedaan program Stack antara menggunakan Array dan Linked List!
2. Jelaskan Aplikasi-Aplikasi Stack dalam dunia nyata!
3. Tuliskan contoh program pada operasi Full!
4. Tuliskan contoh program pada operasi Empty!

TUGAS AKHIR

1. Buatlah program untuk mengetahui suatu kalimat adalah polindrom atau tidak!
Polindrom adalah suatu kata atau kalimat yang jika dibaca dari depan akan sama maknanya dengan jika dibaca dari belakang. Contoh : “KASUR NABABAN RUSAK” maka jika kalimat tersebut dibalik akan mempunyai makna yang sama yaitu: “KASUR NABABAN RUSAK”