



STIKOM BALI

STACK (TUMPUKAN)

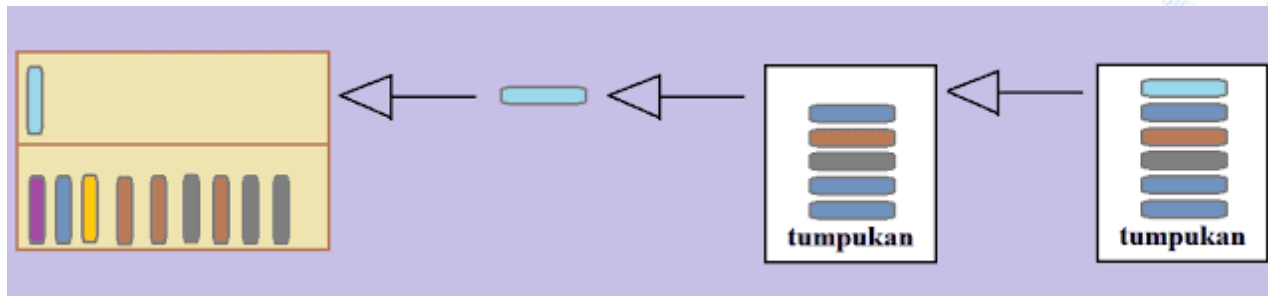
NIKADEK SUMIARI, S.Kom., M.MSI

DEFINISI STACK

- Stack (tumpukan) dapat diartikan sebagai list yang operasi penghapusan dan penyisipannya dapat dilakukan di satu ujung, atau dapat diartikan sebagai suatu kumpulan data yang seolah-olah ada data yang diletakkan diatas data lainnya
- Stack adalah suatu urutan elemen yang elemennya dapat diambil dan ditambah hanya pada posisi akhir (top) saja
- Stack Bersifat LIFO (Last In First Out).

DEFINISI STACK

- Contoh dalam kehidupan sehari-hari adalah tumpukan piring di sebuah restoran yang tumpukannya dapat ditambah pada bagian paling atas dan jika mengambilnya pun dari bagian paling atas pula

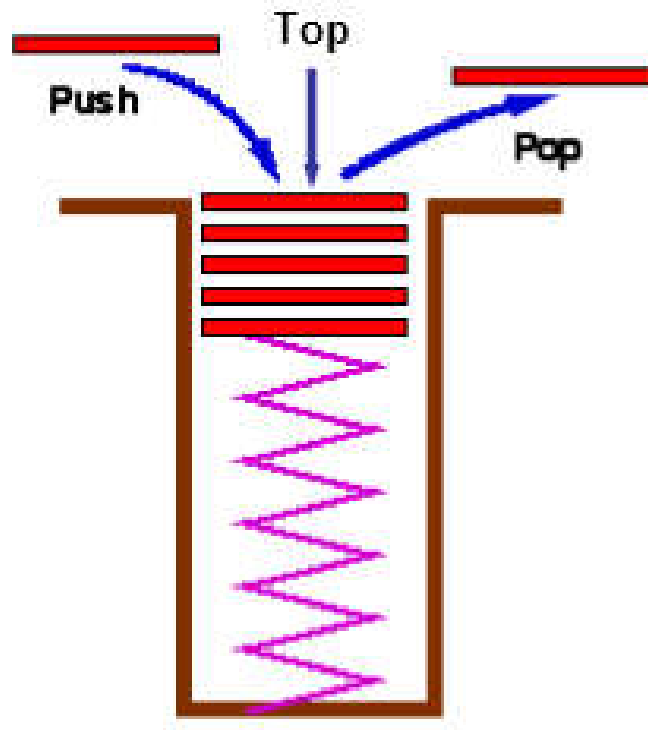


- Seperti gambar diatas tumpukan buku yang pertama kali diambil adalah yang paling atas untuk dipindahkan ke rak

OPERASI PADA STACK

- Push : Menambahkan data pada STACK pada tumpukan paling atas
- Pop : mengambil data pada STACK pada tumpukan paling atas
- IsFull : Mengecek apakah STACK sudah penuh
- IsEmpty : Mengecek apakah STACK sudah kosong
- Clear : untuk mengosongkan stack
- Peek : untuk melihat element teratas dari stack

ILUSTRASI



STIKOM

FUNGSI ISFULL

Untuk memeriksa apakah stack sudah penuh, maka dapat dilakukan dengan cara memeriksa top of stack jika sudah sama dengan MAX_STACK-1 maka full.

Jika belum (masih lebih kecil dari MAX_STACK-1) maka belum full.

Seperti ditunjukkan gambar di berikut ini:

9

Printer



MAX_STACK



Top

8

Komp

7

Oven

6

Mixer

5

Setrika

4

Kulkas

3

DVD

2

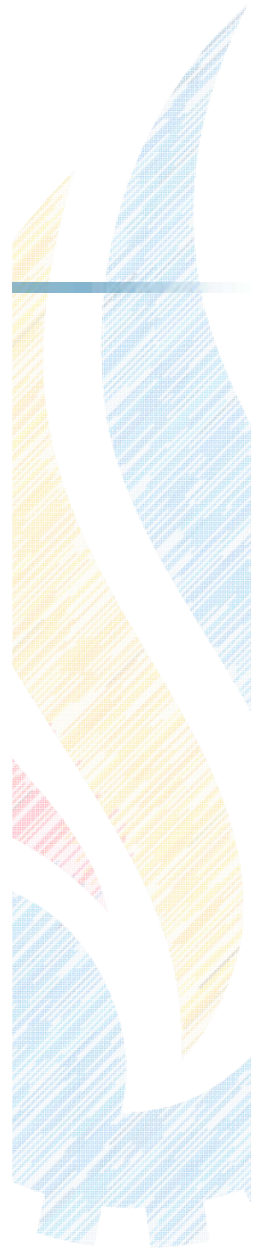
Compo

1

VCD

0

TV

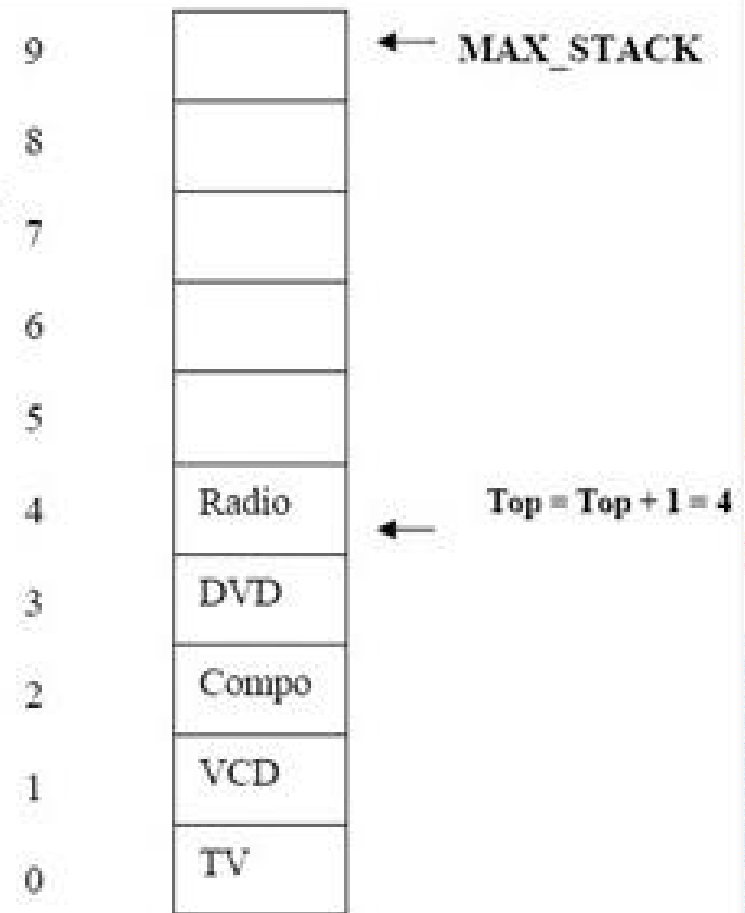
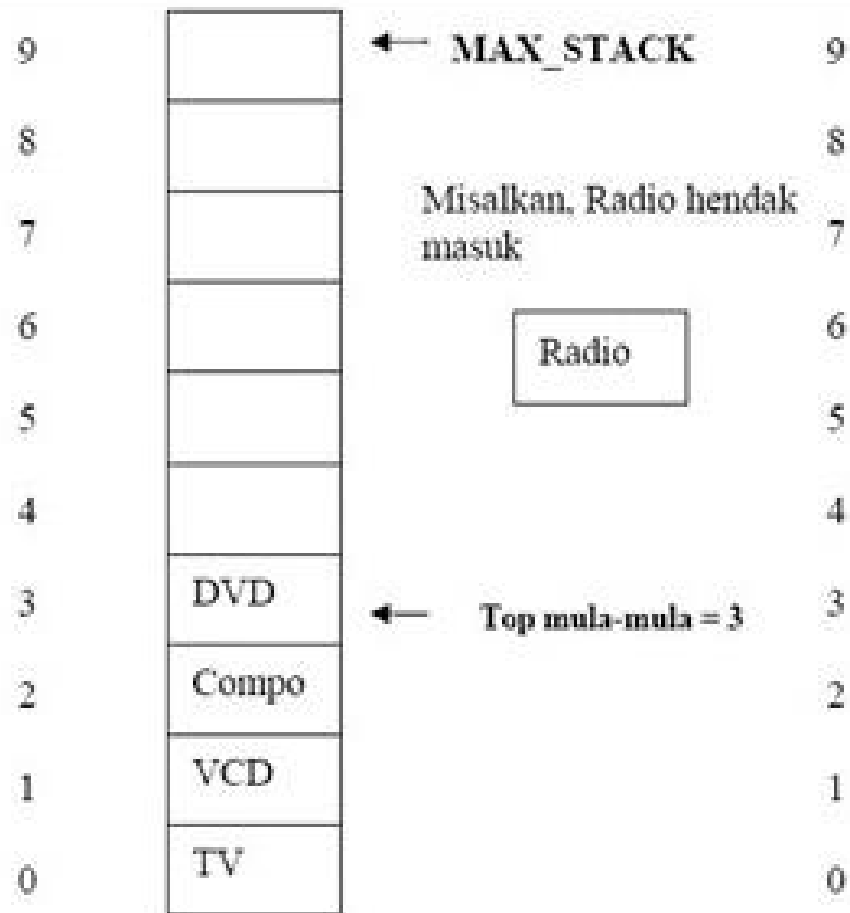


FUNGSI PUSH

Untuk memasukkan elemen ke stack, selalu menjadi elemen teratas stack.

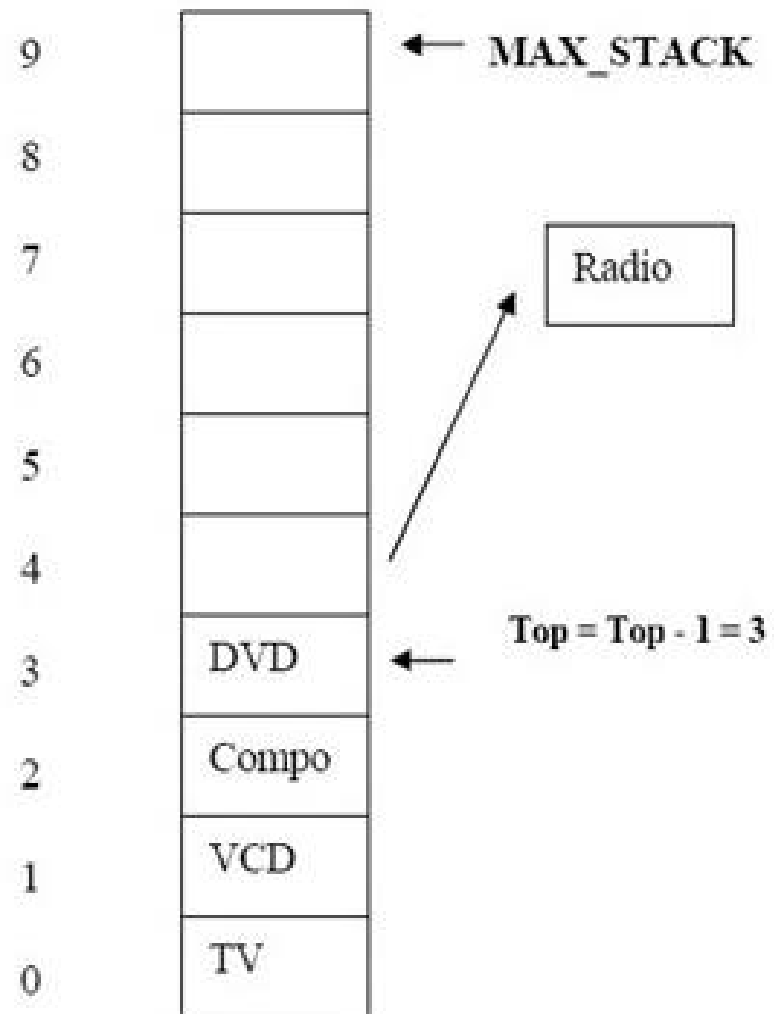
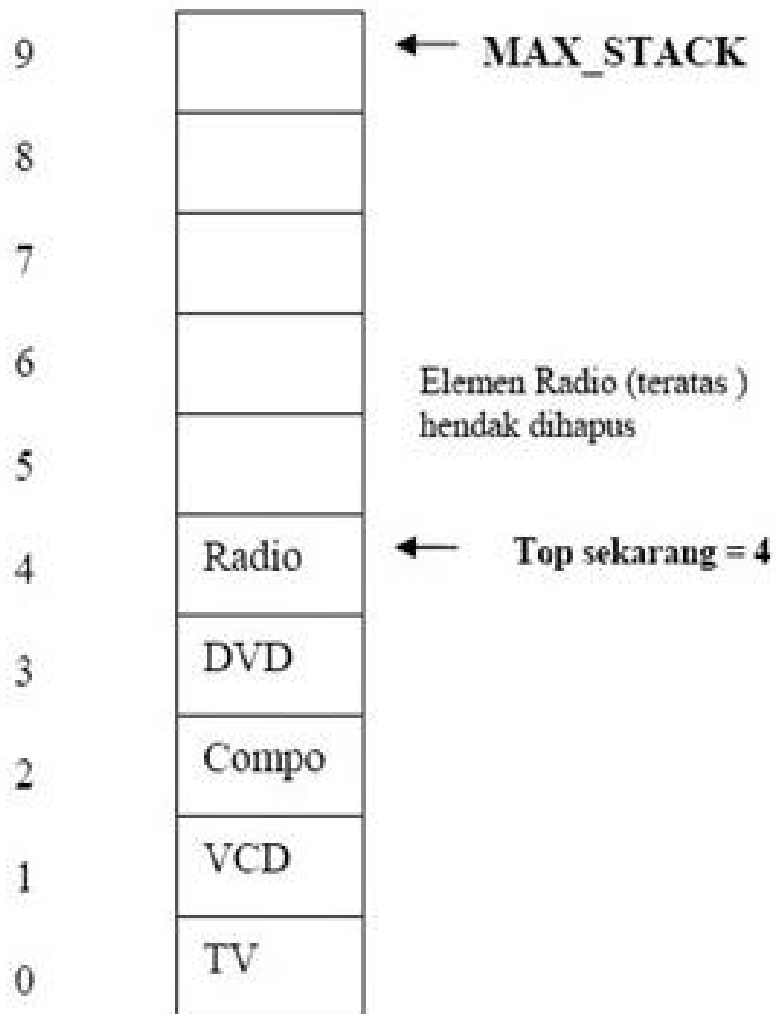
Tambah satu (increment) nilai top of stack terlebih dahulu setiap kali ada penambahan elemen stack, asalkan stack masih belum penuh, kemudian isikan nilai baru ke stack berdasarkan indeks top of stack setelah ditambah satu (diincrement).

Seperti gambar di berikut ini.



FUNGSI POP

Untuk mengambil elemen teratas dari stack. Ambil dahulu nilai elemen teratas stack dengan mengakses top of stack, tampilkan nilai yang akan diambil terlebih dahulu, baru didecrement nilai top of stack sehingga jumlah elemen stack berkurang.



CODE STACK

```
#include <iostream>
#include <stdlib.h>
system("cls")
#include <conio.h>
#define MAX 10
using namespace std;

int top=-1, Stack[MAX];
```


FUNGSI PUSH

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

FUNGSI POP

```
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--];
    }
}
```

FUNGSI CLEAR

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

FUNGSI PRINT

```
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;
    }
}
```


TAMPILAN MENU

```
int main()
{
    int choose;
    do {
        Print();
        cout << "\n1. Push"
              << "\n2. Pop"
              << "\n3. Clear"
              << "\n4. Exit"
              << "\nCodeProgram"
              << "\nPergerakan ( TOP )
: " << top
        << "\n\n>> Choose : ";
        cin >> choose;
        switch (choose){
            case 1:
                push();getch();
                break;
            case 2:
                pop();getch();
                break;
```

```
            case 3:
                clrarr();
                cout << "\nClear Succes" <<
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);
}
```



STIKOM BALI

TERIMA KASIH

STIKOM