

Pertemuan 6

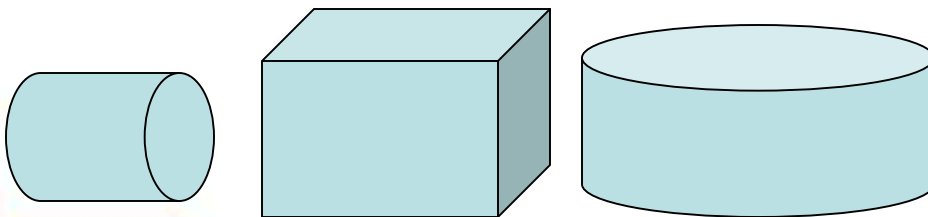
STACK atau TUMPUKAN



STACK (TUMPUKAN)

Merupakan bentuk khusus dari Linier List yang pemasukan dan penghapusan elemennya hanya dapat dilakukan pada satu posisi, yaitu posisi akhir dari List (Top)

Prinsip Stack adalah ***LAST-IN-FIRST-OUT (LIFO)***.



Klik untuk
Ilustrasi Stack

OPERASI STACK

- **ISEMPTY**
Untuk memeriksa apakah stack kosong
- **ISFULL**
Untuk memeriksa apakah stack sudah penuh
- **PUSH**
Untuk menambahkan item pada posisi paling atas (TOP)
- **POP**
Untuk menghapus item paling atas (TOP)
- **CLEAR**
Untuk mengosongkan stack

STACK PADA ARRAY

Deklarasi MAX_STACK

```
#define MAX_STACK 5
```

Deklarasi STACK dengan struct dan array data

```
typedef struct STACK{  
    int top;  
    int data[5];  
};
```

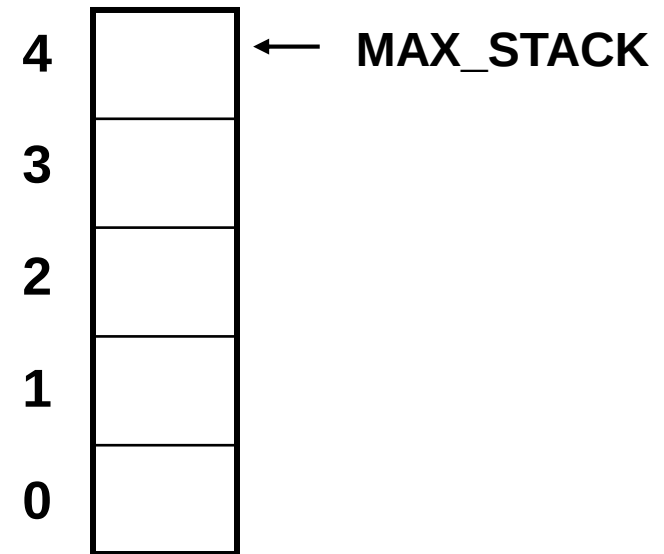
Deklarasi variabel stack dari struct

```
STACK tumpuk;
```

Inisialisasi

- Pada mulanya isi top dengan -1, karena array dalam C/C++ dimulai dari 0, berarti stack adalah KOSONG
- TOP adalah variabel penanda dalam STACK yang menunjukkan elemen teratas Stack.
- TOP of STACK akan selalu bergerak hingga mencapai MAX of STACK sehingga menyebabkan stack PENUH

```
void inisialisasi ()  
{  
    tumpuk.top = -1  
}
```

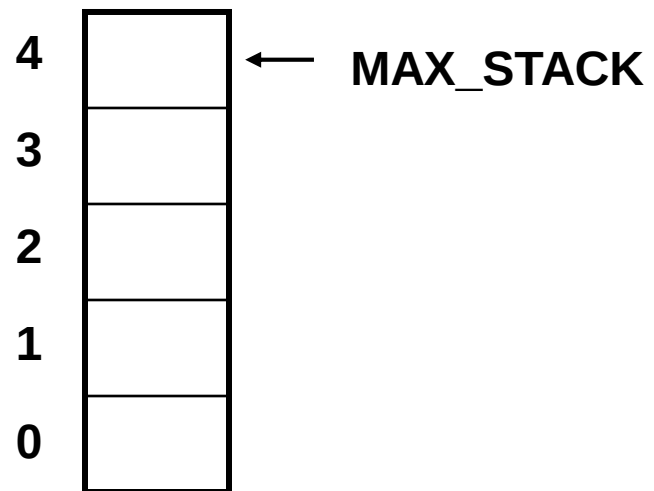


TOP = -1

Fungsi IsEmpty

- Digunakan untuk memeriksa apakah stack masih dalam kondisi kosong
- Dengan cara memeriksa TOP of STACK.
Jika TOP masih = -1 maka berarti stack masih kosong

```
int IsEmpty ()  
{  
    if (tumpuk.top == -1  
        return 1;  
    else  
        return 0;  
}
```



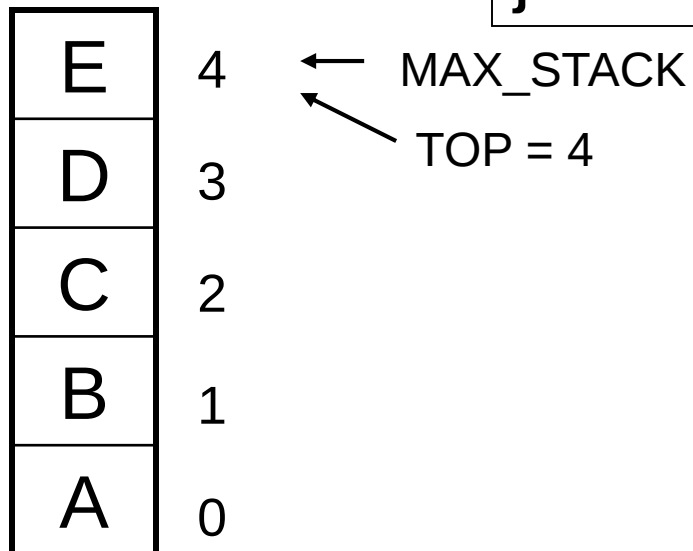
TOP = -1

Fungsi IsFull

- Digunakan untuk memeriksa apakah kondisi stack sudah penuh
- Dengan cara memeriksa TOP of Stack.
Jika $\text{TOP of STACK} = \text{MAX_STACK} - 1$ maka FULL (Penuh).
Jika $\text{TOP of STACK} < \text{MAX_STACK} - 1$ maka belum penuh

Fungsi isFull

```
int IsFull ()  
{  
    if (tumpuk.top == MAX_STACK-  
    1  
        return 1;  
    else  
        return 0;  
}
```

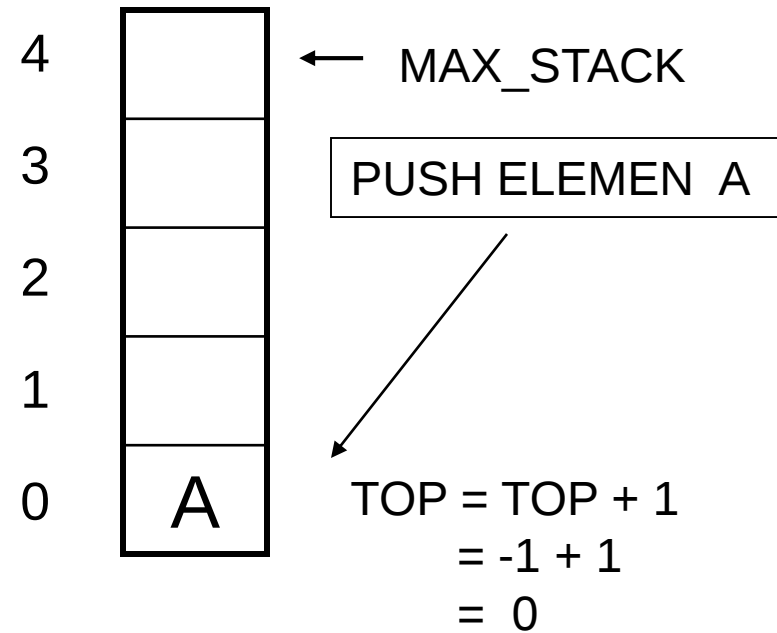
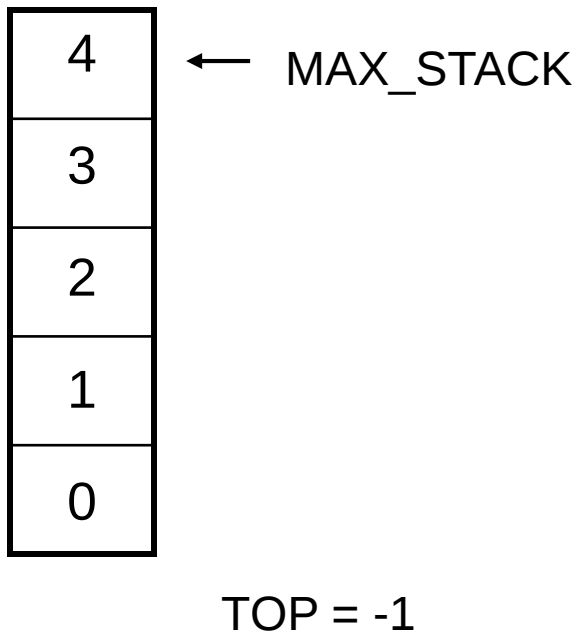


Fungsi PUSH

- Digunakan untuk memasukkan elemen ke dalam stack dan selalu menjadi elemen teratas stack
- Dengan cara :
 1. Menambah satu (increment) nilai TOP of STACK setiap ada penambahan elemen stack selama stack masih belum penuh
 2. Isikan nilai baru ke stack berdasarkan indeks TOP of STACK setelah ditambah satu (diincrement)

Fungsi Push

```
void push (char d[5])  
{  
    tumpuk.top++  
    strcpy(tumpuk.data[tumpuk.top],d);  
}
```

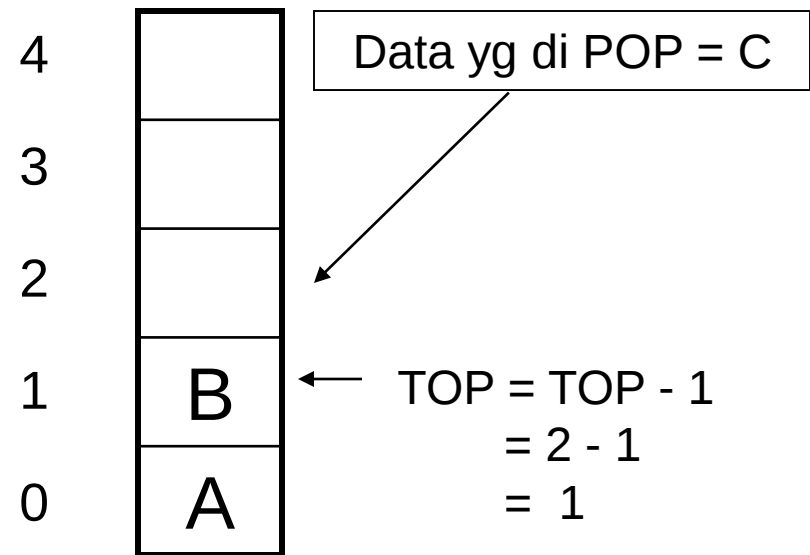
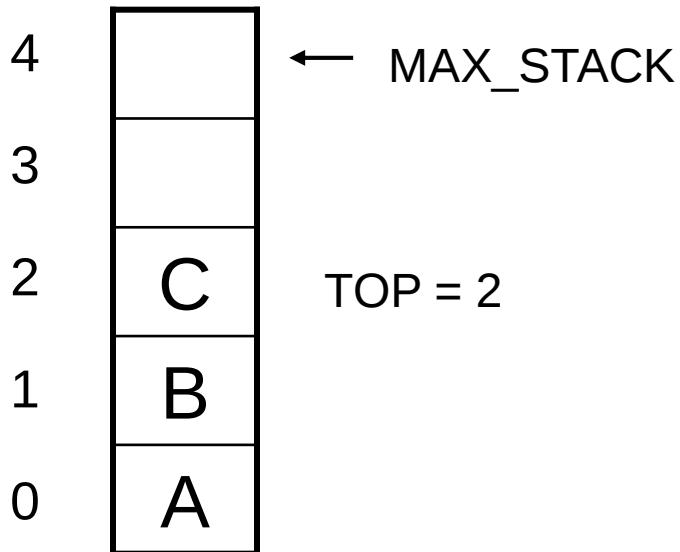


Fungsi POP

- Digunakan untuk menghapus elemen yang berada pada posisi paling atas dari stack.
- Dengan cara :
 1. Ambil dahulu nilai elemen teratas stack dengan mengakses TOP of STACK.
 2. Tampilkan nilai yang akan diambil.
 3. Lakukan decrement nilai TOP of STACK sehingga jumlah elemen stack berkurang 1

Fungsi POP

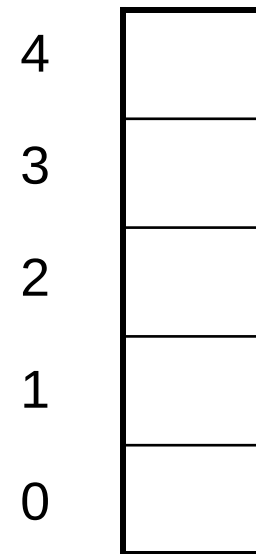
```
void pop ()  
{  
    printf("Data yg di POP = %s\n", tumpuk.data[tumpuk.top]);  
    tumpuk.top--;  
}
```



Fungsi CLEAR

- Digunakan untuk mengosongkan stack / membuat stack hampa sehingga Top pada Stack berada kembali di posisi Top = -1

```
void clear ()  
{  
    tumpuk.data=tumpuk.top=-1  
    printf("Data clear");  
}
```



Latihan I Struktur Data (Pertemuan 6)

Diketahui suatu stack dgn $\text{max_stack} = 6$

1. Bila dilakukan PUSH 3 elemen kedalam stack, kemudian di PUSH lagi 2 elemen dan di POP 3 elemen. Maka dimana posisi Top of Stack ?
2. IsEmpty pada kondisi terakhir adalah ?
3. Dari kondisi diatas, Berapa elemen yg hrs di PUSH unt mencapai kondisi penuh Top of Stack = max_stack ?
4. Berapa elemen yg hrs di POP unt mencapai kondisi IsEmpty = True

**Jawaban dibahas dgn
menggunakan contoh program**

Contoh Program Stack

[klikdisini](#)

Latihan Soal II

Struktur Data (Pertemuan 6)

1. Operasi Stack yang digunakan untuk memeriksa apakah stack sudah penuh, adalah
- a. PUSH
 - b. POP
 - c. CLEAR
 - d. ISEEMPTY
 - e. ISFULL

2. Void push (char d[5])
{
Tumpuk.top++
Strcpy (tumpuk.data [tumpuk.top],d)
}

Merupakan fungsi...

- a. Is Full
- b. TOP
- c. PUSH
- d. POP
- e. CLEAR

2. Void push (char d[5])

{

Tumpuk.top++

Strcpy (tumpuk.data [tumpuk.top],d)

}

Merupakan fungsi...

a. Is Full

d. POP

b. TOP

c. PUSH

e. CLEAR

3. Jika pada stack terdapat kondisi TOP of STACK = MAX_STACK - 1 maka stack berada dalam keadaan ...

a. ISEEMPTY

d. RETREIVE

b. CLEAR

e. FULL

c. TOP

3. Jika pada stack terdapat kondisi TOP of STACK = MAX_STACK - 1 maka stack berada dalam keadaan ...
- a. ISEEMPTY
 - b. CLEAR
 - c. TOP
 - d. RETREIVE
 - e. FULL
4. Prinsip dari stack adalah
- a. First in First Out (FIFO)
 - b. ISFULL
 - c. PUSH
 - d. Last in First Out (LIFO)
 - e. CLEAR

4. Prinsip dari stack adalah

a. First in First Out (FIFO)

b. ISFULL

c. PUSH

d. Last in First Out (LIFO)

e. CLEAR

5. Diberikan perintah/instruksi pada program C++, yaitu **tumpuk.top++**; Maksud dari perintah/instruksi tersebut adalah

a. $\text{Top} = \text{Top} + 1$

b. $\text{Top} = \text{Top} - 1$

c. $\text{Top} = 0$

d. $\text{Top} = - 1$

e. $\text{Top} = 1$

5. Diberikan perintah/instruksi pada program C++, yaitu **tumpuk.top++**; Maksud dari perintah/instruksi tersebut adalah
- a. $\text{Top} = \text{Top} + 1$
 - b. $\text{Top} = \text{Top} - 1$
 - c. $\text{Top} = 0$
 - d. $\text{Top} = - 1$
 - e. $\text{Top} = 1$
1. Operasi Stack yang digunakan untuk memeriksa apakah stack sudah penuh, adalah
- a. PUSH
 - b. POP
 - c. CLEAR
 - d. ISEEMPTY
 - e. ISFULL