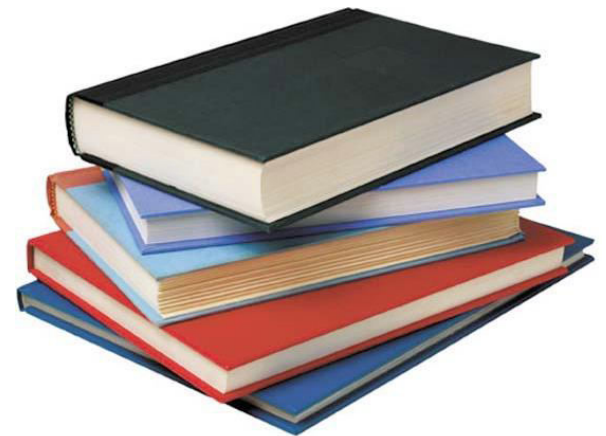


Tumpukan (Stack)

Farah Zakiyah Rahmanti

2014

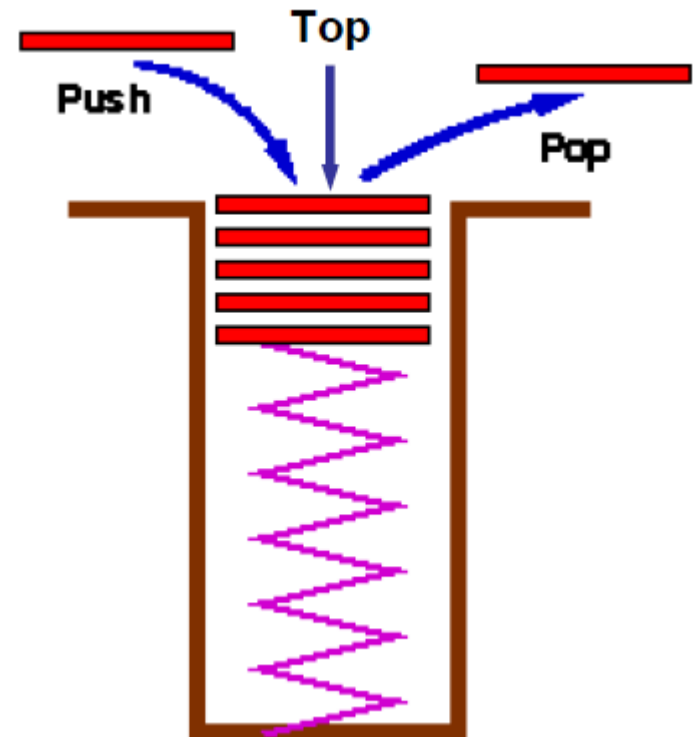


Deskripsi

- konsep yang efektif untuk menyimpan dan mengambil data dengan aturan "terakhir masuk sebagai yang pertama keluar (Last in First Out / LIFO)".
- pengambilan data akan berkebalikan urutannya dengan penyimpanan data.

Push ()

- Operasi push adalah operasi untuk menambahkan sebuah elemen dan diletakkan pada posisi paling atas dari sebuah tumpukan.



Contoh push ()

```
struct t_myStack {  
    int data[100];  
    int amount;  
};  
  
typedef struct t_myStack myStack;  
  
myStack Stacksss;
```

```
void push(myStack* s, int myData)  
{  
    s->data[s->amount] = myData;  
    s->amount++;  
}
```

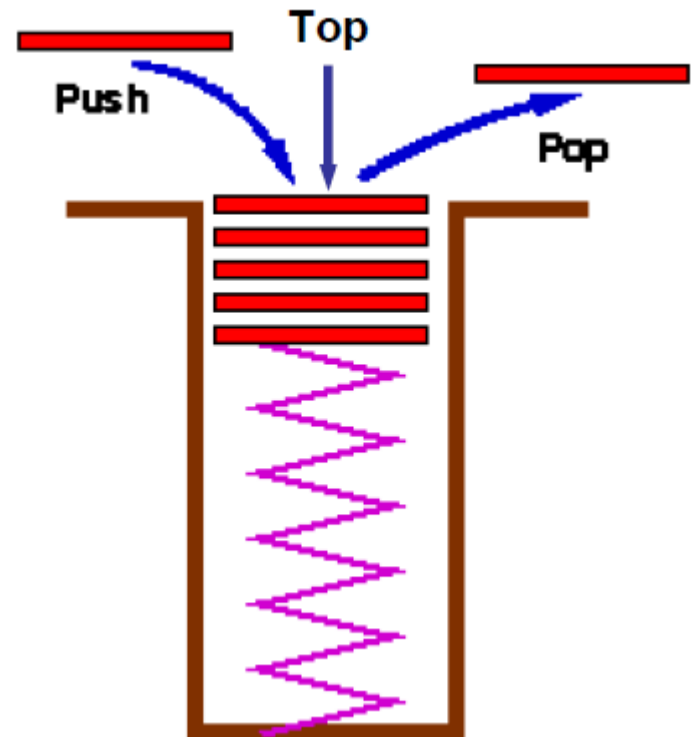
procedure **push()**

```
push(&Stacksss, 10);  
push(&Stacksss, 20);  
push(&Stacksss, 30);
```

Pemanggilan procedure **push()** dari **main()**

Pop ()

- Operasi pop adalah operasi untuk mengambil elemen yang terletak pada posisi paling atas dari sebuah tumpukan.



Contoh pop ()

```
struct t_myStack {  
    int data[100];  
    int amount;  
};  
  
typedef struct t_myStack myStack;  
  
myStack Stacksss;
```

```
int pop(myStack* s)  
{  
    s->amount--;  
    return s->data[s->amount];  
}
```

Function **pop()**

```
for(i=0; Stacksss.amount>0; i++)  
    printf("%d ", pop(&Stacksss));
```

Pemanggilan function **pop()** dari **main()**