

Praktikum Algoritma & Struktur Data  
**Tugas 2: Implementasi stack**

**Dosen Pengampu**

Dr. Tita Karlita S.Kom, M.Kom



**Disusun Oleh :**

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA  
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA  
TAHUN AKADEMIK 2021/2022**

## Tugas 2: Implementasi stack

Deskripsi :

Implementasikan stack dengan operasi: 1. Push 2. Pop 3. Tampil

Batas Waktu :

Sabtu, 26 Maret 2022 - 01:05

Source Code:

```
#include <stdio.h>
#include <stdlib.h>
#define MAX 5 // Jumlah maksimum elemen yang dapat disimpan

int top, stack[MAX];
void push();
void pop();
void display();

void main()
{
    int ch;
    while (1) // loop tak terbatas, akan berakhir ketika pilihannya
    adalah 4
    {
        printf("Menu Stack Using Array\n");
        printf("1. Mengisi Stack (Push)\n");
        printf("2. Mengambil Isi Stack (Pop)\n");
        printf("3. Menampilkan Isi Stack -> LIFO \n");
        printf("4. Keluar \n");
        printf("\nMasukkan Pilihan: ");
        scanf("%d", &ch);
        switch (ch)
        {
            case 1: // Mengisi Stack
                push();
                break;
            case 2: // Mengambil Isi Stack
                pop();
                break;
            case 3: // Menampilkan Isi Stack
                display();
                break;
            case 4: // Keluar
                exit(0);
            default:
                printf("\nPilihan Tidak Ada !");
        }
    }
}

void push() // Mengisi Stack
```

```
{  
    char x;  
    if (top == MAX)  
    {  
        printf("\nStack Penuh");  
    }  
    else  
    {  
        printf("\nMasukkan data Anda : ");  
        scanf("%s", &x);  
        top++;  
        stack[top] = x;  
    }  
}  
  
void pop() // Mengambil Isi Stack  
{  
    if (top == 0)  
    {  
        printf("\nStack Kosong");  
    }  
    else  
    {  
        printf("\n%c Berhasil dihapus\n", stack[top]);  
        top--;  
    }  
}  
  
void display() // Menampilkan Isi Stack  
{  
    int i;  
    if (top == 0)  
    {  
        printf("\nStack Kosong");  
    }  
    else  
    {  
        printf("\nIsi dari Stack\n");  
        for (i = top; i >= 1; --i)  
            printf("%c\n", stack[i]);  
    }  
}
```

### Output Source Code :

```
"G:\My Drive\COLLAGE\PENS\SEMESTER 2\Algoritma & Struktur Data\stack\kode.exe"
Masukkan Pilihan: 1

Masukkan data Anda : b
Menu Stack Using Array
1. Mengisi Stack (Push)
2. Mengambil Isi Stack (Pop)
3. Menampilkan Isi Stack -> LIFO
4. Keluar

Masukkan Pilihan: 1

Masukkan data Anda : c
Menu Stack Using Array
1. Mengisi Stack (Push)
2. Mengambil Isi Stack (Pop)
3. Menampilkan Isi Stack -> LIFO
4. Keluar

Masukkan Pilihan: 2

c Berhasil dihapus
Menu Stack Using Array
1. Mengisi Stack (Push)
2. Mengambil Isi Stack (Pop)
3. Menampilkan Isi Stack -> LIFO
4. Keluar

Masukkan Pilihan: 3

Isi dari Stack
b
a
Menu Stack Using Array
1. Mengisi Stack (Push)
2. Mengambil Isi Stack (Pop)
3. Menampilkan Isi Stack -> LIFO
4. Keluar

Masukkan Pilihan: A
```

Github : <https://github.com/FazaZas/algoritma-struktur-data-stack.git>