

# PROGRAM GUIDE PENGAYAAN MATA KULIAH STRUKTUR DATA WEEK 9

Nama : Mokhamad Fadlika Maulana  
N I M : 20230801194

Program di atas adalah program pengelolaan tumpukan buku menggunakan bahasa C++. Program ini memungkinkan pengguna untuk melakukan beberapa operasi dasar pada tumpukan buku seperti menambah buku, mengambil buku, melihat buku paling atas, mengecek apakah tumpukan kosong, dan melihat jumlah buku dalam tumpukan. Program ini memanfaatkan struktur data stack (tumpukan) yang disediakan oleh pustaka standar C++ (<stack>).

## Tujuan Program

Tujuan dari program ini adalah untuk memberikan contoh implementasi dari struktur data tumpukan (stack) dan operasi-operasi yang dapat dilakukan pada tumpukan tersebut. Program ini juga bertujuan untuk mempraktikkan penggunaan fungsi, pengendalian alur program dengan while loop, dan pemilihan aksi dengan switch case.

## Penjelasan Fungsi-Fungsi

### ➤ clearScreen():

Deskripsi: Fungsi ini digunakan untuk membersihkan layar tergantung pada sistem operasi yang digunakan. Pada Windows, ia menggunakan perintah cls, sementara pada Unix/Linux menggunakan perintah clear.

Tujuan: Membuat tampilan program lebih rapi dengan membersihkan layar sebelum menampilkan menu.

### ➤ tambahBuku(stack<string>& tumpukanBuku):

Deskripsi: Fungsi ini meminta pengguna memasukkan nama buku yang ingin ditambahkan ke dalam tumpukan. Nama buku tersebut kemudian ditambahkan ke tumpukan.

Tujuan: Menambah elemen (buku) ke dalam tumpukan.

### ➤ ambilBuku(stack<string>& tumpukanBuku):

Deskripsi: Fungsi ini mengambil buku dari tumpukan (elemen paling atas) dan mengeluarkannya dari tumpukan. Jika tumpukan kosong, ia akan memberikan pesan bahwa tumpukan kosong.

Tujuan: Mengeluarkan elemen (buku) dari tumpukan.

➤ `lihatBukuPalingAtas(const stack<string>& tumpukanBuku):`

Deskripsi: Fungsi ini menampilkan buku yang berada di paling atas tumpukan tanpa mengeluarkannya. Jika tumpukan kosong, ia akan memberikan pesan bahwa tumpukan kosong.

Tujuan: Melihat elemen (buku) paling atas dari tumpukan tanpa mengeluarkannya.

➤ `cekApakahTumpukanKosong(const stack<string>& tumpukanBuku):`

Deskripsi: Fungsi ini memeriksa apakah tumpukan kosong atau tidak, dan memberikan pesan yang sesuai.

Tujuan: Mengetahui status kosong atau tidaknya tumpukan

➤ `lihatJumlahBuku(const stack<string>& tumpukanBuku):`

Deskripsi: Fungsi ini menampilkan jumlah buku yang ada dalam tumpukan.

Tujuan: Mengetahui jumlah elemen (buku) dalam tumpukan.

## Deskripsi dan Tujuan Bagian Utama Program

- `stack<string> tumpukanBuku:` Mendeklarasikan tumpukan buku menggunakan template stack dari pustaka standar C++.
- `int pilihan:` Variabel untuk menyimpan pilihan pengguna dari menu.
- `bool keluar = false:` Variabel boolean untuk menentukan apakah program harus keluar dari loop.
- `while (!keluar):` Loop utama yang terus berjalan hingga pengguna memilih untuk keluar dari program.
- `clearScreen():` Memanggil fungsi untuk membersihkan layar sebelum menampilkan menu.
- `cout << ...:` Menampilkan menu dan meminta pengguna untuk memasukkan pilihan.
- `cin >> pilihan:` Menerima input pilihan dari pengguna.
- `cin.ignore():` Mengabaikan karakter newline setelah input pilihan agar tidak mengganggu input berikutnya.
- `switch (pilihan):` Memilih aksi berdasarkan input pengguna.
- `case 1:` Memanggil fungsi `tambahBuku(tumpukanBuku)`.
- `case 2:` Memanggil fungsi `ambilBuku(tumpukanBuku)`.
- `case 3:` Memanggil fungsi `lihatBukuPalingAtas(tumpukanBuku)`.
- `case 4:` Memanggil fungsi `cekApakahTumpukanKosong(tumpukanBuku)`.
- `case 5:` Memanggil fungsi `lihatJumlahBuku(tumpukanBuku)`.
- `case 6:` Mengubah variabel `keluar` menjadi `true` untuk keluar dari loop.
- `default:` Menampilkan pesan kesalahan jika pilihan tidak valid.
- `if (!keluar):` Jika belum keluar, meminta pengguna menekan Enter untuk melanjutkan.
- `if (cin.fail()) :` Blok kode ini menangani kasus di mana pengguna memasukkan input yang tidak valid (seperti memasukkan huruf ketika diharapkan angka).
- `#include <iostream>:` Untuk operasi input dan output.
- `#include <stack>:` Untuk menggunakan struktur data stack.

- `#include <string>`: Untuk manipulasi string.
- `#include <limits>`: Untuk mendapatkan batas numerik, terutama digunakan dalam mengabaikan input yang tidak valid.

## **Kesimpulan**

Program ini menunjukkan bagaimana menggunakan struktur data stack untuk mengelola data dengan operasi dasar seperti push, pop, peek, cek kosong, dan cek ukuran. Penggunaan fungsi-fungsi memudahkan dalam memisahkan logika program dan membuat kode lebih modular dan mudah dibaca. Program ini juga memperkenalkan penggunaan perintah sistem untuk membersihkan layar, yang membantu dalam membuat antarmuka pengguna lebih bersih dan mudah digunakan.