

Nama : Aditya Irfan Hadzami

NIM : 2209076020

Penjelasan Soal Nomor 1

1. Program ini adalah untuk mempermudah pengelolaan data mahasiswa secara efisien dan terstruktur. Dengan menggunakan program ini, pengguna dapat menyimpan, menambah, menghapus, menampilkan, dan mengurutkan data mahasiswa berdasarkan IPK secara cepat tanpa harus melakukannya secara manual. Program ini juga melatih penggunaan konsep-konsep dasar pemrograman seperti struct, penggunaan array of pointers, dan manajemen memori, yang sangat penting dalam pengelolaan data berbasis memori dinamis. Selain itu, implementasi algoritma sorting memberikan pengalaman dalam mengorganisir data agar mudah diakses sesuai dengan kriteria tertentu. Program ini bermanfaat bagi pengelola data akademik atau sistem administrasi kampus yang ingin menjaga data mahasiswa tetap terstruktur dan mudah dikelola.

Adapun penjelasan tentang codingan program ini adalah

Program ini dirancang untuk mengelola data mahasiswa menggunakan struct, array of pointers, dan beberapa fungsi yang berhubungan. Program ini menggunakan struct bernama Mahasiswa yang berisi NIM, nama, dan IPK untuk menyimpan informasi mahasiswa. Data disimpan dalam sebuah array of pointers yang memungkinkan penyimpanan hingga 10 data mahasiswa. Program ini menyediakan beberapa fitur utama, yaitu menambah, menghapus, menampilkan, dan mengurutkan data mahasiswa berdasarkan IPK. Fungsi tambahMahasiswa() digunakan untuk menambahkan data baru, sedangkan hapusMahasiswa() menghapus data berdasarkan NIM yang diberikan oleh pengguna. Data mahasiswa dapat dilihat menggunakan tampilkanMahasiswa(), dan diurutkan secara descending berdasarkan IPK dengan algoritma Bubble Sort yang sederhana dalam fungsi urutkanMahasiswa(). Program ini juga memiliki manajemen memori manual menggunakan new dan delete untuk mencegah kebocoran memori. Menu interaktif memudahkan pengguna untuk memilih tindakan yang diinginkan, seperti menambah atau menghapus data, hingga program diakhiri.

## Penjelasan Soal Nomor 2

2. Program ini adalah untuk mempermudah pengelolaan inventaris peralatan laboratorium, yang memungkinkan pengguna untuk menyimpan, mengubah, menghapus, dan menampilkan data secara terstruktur dan efisien. Dengan adanya file handling, data inventaris dapat disimpan secara permanen dan diakses kembali saat diperlukan, sehingga mengurangi kemungkinan kehilangan data. Program ini juga memberikan kemudahan dalam melacak kondisi dan jumlah peralatan, membantu pengelola laboratorium untuk melakukan pengawasan dan pemeliharaan secara lebih baik. Selain itu, program ini juga berfungsi sebagai sarana belajar dalam penerapan konsep struct, manipulasi data dengan vector, dan pemrograman file di C++.

Adapun penjelasan tentang codingan program ini adalah

Program ini dirancang untuk mengelola inventaris peralatan laboratorium menggunakan struct dan file handling. Berikut adalah penjelasan tentang bagian-bagian utama dari program ini:

**Struct Peralatan:** Struct ini digunakan untuk menyimpan informasi mengenai peralatan laboratorium, dengan anggota:

- kode (string): Kode unik untuk setiap peralatan.
- nama (string): Nama dari peralatan.
- jumlah (int): Jumlah unit peralatan yang tersedia.
- kondisi (string): Deskripsi mengenai kondisi peralatan (misalnya, baik, rusak, perlu perawatan).

**Fungsi untuk Mengelola Data:**

- **tambahPeralatan:** Mengizinkan pengguna untuk menambahkan data peralatan baru ke dalam inventaris.
- **ubahPeralatan:** Mengizinkan pengguna untuk mengubah data peralatan berdasarkan kode yang dimasukkan. Jika kode ditemukan, data peralatan akan diperbarui.
- **hapusPeralatan:** Menghapus data peralatan berdasarkan kode. Jika kode tidak ditemukan, program akan memberitahu pengguna bahwa peralatan tersebut tidak ada.

**Fungsi File Handling:**

- **simpanData:** Menyimpan semua data peralatan yang ada ke dalam file teks bernama "inventaris.txt". Setiap peralatan disimpan dengan informasi di baris terpisah.
- **bacaData:** Membaca data peralatan dari file teks dan menambahkannya ke dalam inventaris. Fungsi ini memperhatikan format penyimpanan sehingga data dapat dibaca dengan benar.

**Menampilkan Laporan:**

- **tampilkanLaporan:** Menampilkan laporan inventaris yang diurutkan berdasarkan kode peralatan. Data ditampilkan dalam format tabel yang memudahkan pengguna untuk membaca informasi.

**Menu Interaktif:**

- Program memiliki menu interaktif yang memungkinkan pengguna untuk memilih opsi yang diinginkan. Pengguna dapat menambah, mengubah, menghapus data, menyimpan dan membaca data dari file, serta menampilkan laporan inventaris.

### Penjelasan Soal Nomor 3

3. Program ini adalah untuk memberikan alat yang sederhana namun efektif untuk mengevaluasi ekspresi aritmatika dalam notasi postfix. Dengan menggunakan struktur data stack, program ini dapat menangani operasi aritmatika dasar serta pemangkatan dengan cara yang efisien. Langkah-langkah evaluasi ditampilkan untuk membantu pengguna memahami proses evaluasi dan logika di balik penggunaan notasi postfix. Program ini juga berguna sebagai latihan dalam penerapan struktur data dan pengolahan string di dalam bahasa pemrograman C++.

Adapun penjelasan tentang codingan program ini adalah

Program ini merupakan kalkulator yang menggunakan notasi postfix (Reverse Polish Notation - RPN) untuk melakukan evaluasi ekspresi aritmatika. Berikut adalah penjelasan tentang bagian-bagian utama program ini:

#### 1. **Kelas Stack:**

- Kelas ini digunakan untuk mengimplementasikan struktur data stack.
- Memiliki metode push untuk menambahkan elemen ke dalam stack, pop untuk mengambil dan menghapus elemen teratas dari stack, dan isEmpty untuk memeriksa apakah stack kosong.

#### 2. **Fungsi evaluatePostfix:**

- Fungsi ini menerima ekspresi postfix sebagai string dan mengevaluasinya menggunakan stack.
- Menggunakan istringstream untuk memecah string ekspresi menjadi token (angka dan operator).
- Jika token adalah angka, ia akan diubah menjadi double dan dimasukkan ke dalam stack. Jika token adalah operator, dua operand diambil dari stack untuk melakukan operasi aritmatika yang sesuai (penjumlahan, pengurangan, perkalian, pembagian, dan pemangkatan).
- Hasil dari operasi aritmatika juga dimasukkan kembali ke dalam stack, dan setiap langkah evaluasi dicetak ke layar untuk memberikan informasi tentang proses.

### 3. **Fungsi main:**

- Fungsi ini meminta pengguna untuk memasukkan ekspresi postfix yang akan dievaluasi.
- Ekspresi harus dipisahkan dengan spasi (misalnya:  $5\ 2\ +\ 3\ ^$ ).
- Hasil evaluasi akan ditampilkan, serta langkah-langkah yang diambil selama evaluasi.

#### Penjelasan Soal Nomor 4

4. Program ini adalah untuk memberikan gambaran tentang bagaimana antrian layanan pelanggan di sebuah bank dapat diatur menggunakan struktur data queue. Dengan simulasi ini, pengguna dapat melihat bagaimana pelanggan dilayani secara bersamaan di beberapa loket, serta menghitung waktu tunggu rata-rata dan statistik lainnya. Program ini juga bertujuan untuk mengajarkan konsep dasar penggunaan queue dalam pemrograman, serta memberikan pemahaman tentang bagaimana proses antrian dapat dimodelkan secara efektif.

Adapun penjelasan tentang coding program ini adalah

Program ini mensimulasikan antrian layanan pelanggan di sebuah bank dengan menggunakan struktur data queue. Berikut adalah penjelasan tentang bagian-bagian utama dari program ini:

1. **Struct Pelanggan:**

- Struct ini digunakan untuk menyimpan informasi mengenai pelanggan yang sedang antri. Terdapat dua anggota:
  - nomorAntrian: Nomor urut pelanggan di antrian.
  - waktuLayanan: Waktu yang diperlukan untuk melayani pelanggan dalam satuan menit.

2. **Kelas Queue:**

- Kelas ini digunakan untuk mengelola antrian pelanggan dengan menggunakan `std::queue` dari C++ STL.
- Memiliki metode `enqueue` untuk menambahkan pelanggan ke dalam antrian, `dequeue` untuk mengeluarkan pelanggan dari antrian, `isEmpty` untuk memeriksa apakah antrian kosong, dan `size` untuk mendapatkan jumlah pelanggan dalam antrian.

3. **Fungsi simulasiAntrian:**

- Fungsi ini mensimulasikan proses antrian layanan pelanggan.
- Pertama, pelanggan ditambahkan ke dalam antrian dengan nomor urut dan waktu layanan acak (1-10 menit).

- Kemudian, fungsi ini mensimulasikan pelayanan pelanggan di tiga loket. Setiap loket akan mengambil pelanggan dari antrian dan menghitung waktu tunggu serta waktu layanan.
- Total waktu tunggu dihitung dan disimpan dalam sebuah variabel. Setelah semua pelanggan dilayani, fungsi ini menghitung rata-rata waktu tunggu dan menampilkan statistik.

#### 4. **Fungsi main:**

- Fungsi ini meminta pengguna untuk memasukkan jumlah pelanggan yang akan disimulasikan.
- Setelah itu, fungsi simulasiAntrian dipanggil untuk menjalankan simulasi.

## Penjelasan Soal Nomor 5

5. Program manajemen perpustakaan ini adalah untuk menyediakan solusi yang efisien dan terorganisir dalam pengelolaan koleksi buku dan proses peminjaman di perpustakaan. Dengan memanfaatkan berbagai struktur data seperti array, pointer, stack, dan queue, program ini memungkinkan pengguna untuk melakukan sejumlah fungsi penting, termasuk penambahan, pencarian, dan tampilan data buku yang tersimpan. Selain itu, dengan menggunakan queue untuk mengelola antrian peminjaman, program ini menciptakan sistem yang adil dan efisien untuk memproses peminjaman buku sesuai urutan, sehingga setiap pelanggan dapat dilayani dengan baik.

Adapun penjelasan tentang codingan program ini adalah

Program ini merupakan sistem manajemen perpustakaan sederhana yang mengelola data buku, riwayat peminjaman, dan antrian peminjaman menggunakan konsep pemrograman berorientasi objek, termasuk penggunaan struct, array of pointers, stack, dan queue. Berikut adalah penjelasan mengenai bagian-bagian utama dari program ini:

1. **Struct Buku:**

- Struct ini digunakan untuk menyimpan informasi mengenai buku di perpustakaan, termasuk ISBN, judul, pengarang, dan tahun terbit.

2. **Kelas Perpustakaan:**

- Kelas ini merupakan pusat manajemen perpustakaan yang menyimpan array of pointers ke struct Buku, stack untuk menyimpan riwayat peminjaman, dan queue untuk mengelola antrian peminjaman.
- Metode dalam kelas ini mencakup fungsi untuk menambah, mencari, dan menampilkan data buku, meminjam dan mengembalikan buku, serta mengelola proses antrian peminjaman.

3. **Fungsi untuk Menambah Buku:**

- tambahBuku: Menambah buku baru ke dalam koleksi jika kapasitas maksimum belum tercapai.



**4. Fungsi untuk Mencari Buku:**

- cariBuku: Mencari buku berdasarkan ISBN dan mengembalikan pointer ke buku jika ditemukan.

**5. Fungsi untuk Menampilkan Buku:**

- tampilkanBuku: Menampilkan semua buku yang ada dalam koleksi.

**6. Fungsi untuk Meminjam dan Mengembalikan Buku:**

- pinjamBuku: Mengelola antrian peminjaman dengan menambahkan buku ke dalam queue.
- kembalikanBuku: Mengembalikan buku terakhir yang dipinjam dan menghapusnya dari stack riwayat peminjaman.

**7. Fungsi untuk Memproses Antrian:**

- prosesAntrian: Memproses semua buku dalam antrian peminjaman dan menambahkannya ke dalam riwayat peminjaman.

**8. Fungsi untuk Menampilkan Riwayat Peminjaman:**

- tampilkanRiwayatPeminjaman: Menampilkan semua buku yang telah dipinjam