

Soal Post Test
Mata Kuliah Struktur Data dan Algoritma
IF-11-C

Aturan dalam mengerjakan:

1. Dalam mengerjakan HANYA diperbolehkan untuk membuka modul.(TIDAK BOLEH membuka internet, browser, GPT, dan media lain selain modul).
2. Tidak boleh kerjasama dengan siapapun.
3. Jawaban yang sama persis akan dianggap jawaban kerjasama dan copy paste (nilai akan dibagi sejumlah jawaban yang sama).
4. Dilarang menggunakan media komunikasi apapun.
5. Waktu penggerjaan 90 menit.
6. Dikerjakan di Ms Word (diketik).

Setiap jawaban wajib menyertakan **source code, penjelasan dan screenshoot hasilnya.**

Tugas: Program Kasir Toko Alat Tulis

Buatlah program kasir sederhana untuk toko alat tulis. Program ini harus memungkinkan user untuk menambahkan, mengubah, dan menghapus data barang. Gunakan minimal 4 materi dari 9 materi modul yang telah dipelajari.

Kasus yang harus diimplementasikan dalam program:

1. Menambahkan 6 jenis barang.
2. Mengubah kode barang.
3. Mengubah jumlah barang.
4. Menghapus 2 jenis barang.

Tampilan menu :

Toko Alat Tulis Atmajaya

- 1. Tambah Data**
- 2. Hapus Data**

- 3. Update Data**
- 4. Tambah Data Urutan Tertentu**
- 5. Hapus Data Urutan Tertentu**
- 6. Hapus Seluruh Data**
- 7. Cari data di posisi tertentu**
- 8. Tampilkan Data**
- 9. Exit**

Contoh Tampilan akhir

Kode barang	Nama Barang	Jumlah Barang	Harga Barang
A01	Buku Gambar	25	4000
A02	Buku Tulis	30	3500
B03	Pulpen Joyko	35	3000
B05	Pensil Joyko	40	2500

Source Code

```
//Muhammad Hamzah Haifan M
//2311102091
//IF-11-C
#include <iostream>
#include <string>
using namespace std;

struct Barang {
    string kode;
    string nama;
    int jumlah;
    double harga;
};

const int MAX_BARANG = 6;
Barang daftarBarang[MAX_BARANG];
int jumlahBarang = 0;

void tambahData() {
    if (jumlahBarang < MAX_BARANG) {
        cout << "Masukkan kode barang: ";
        cin >> daftarBarang[jumlahBarang].kode;
        cout << "Masukkan nama barang: ";
        cin >> daftarBarang[jumlahBarang].nama;
        cout << "Masukkan harga barang: ";
        cin >> daftarBarang[jumlahBarang].harga;
        cout << "Masukkan jumlah barang: ";
        cin >> daftarBarang[jumlahBarang].jumlah;
        jumlahBarang++;
        cout << "Data barang berhasil ditambahkan." << endl;
    } else {
        cout << "Maaf, kapasitas maksimum sudah tercapai." << endl;
    }
}

void hapusData() {
    string kodeHapus;
    cout << "Masukkan kode barang yang ingin dihapus: ";
    cin >> kodeHapus;
    for (int i = 0; i < jumlahBarang; i++) {
        if (daftarBarang[i].kode == kodeHapus) {
            for (int j = i; j < jumlahBarang - 1; j++) {
                daftarBarang[j] = daftarBarang[j + 1];
            }
            jumlahBarang--;
            cout << "Data barang berhasil dihapus." << endl;
        }
    }
}
```

```

        return;
    }
}
cout << "Data barang tidak ditemukan." << endl;
}

void updateData() {
    string kodeUpdate;
    cout << "Masukkan kode barang yang ingin diupdate: ";
    cin >> kodeUpdate;
    for (int i = 0; i < jumlahBarang; i++) {
        if (daftarBarang[i].kode == kodeUpdate) {
            cout << "Masukkan harga baru: ";
            cin >> daftarBarang[i].harga;
            cout << "Masukkan jumlah baru: ";
            cin >> daftarBarang[i].jumlah;
            cout << "Data barang berhasil diupdate." << endl;
            return;
        }
    }
    cout << "Data barang tidak ditemukan." << endl;
}

void hapusDataUrutanTertentu() {
    int indexHapus;
    cout << "Masukkan indeks barang yang ingin dihapus: ";
    cin >> indexHapus;
    if (indexHapus >= 0 && indexHapus < jumlahBarang) {
        for (int i = indexHapus; i < jumlahBarang - 1; i++) {
            daftarBarang[i] = daftarBarang[i + 1];
        }
        jumlahBarang--;
        cout << "Data barang pada indeks " << indexHapus << " berhasil
dihapus." << endl;
    } else {
        cout << "Indeks barang tidak valid." << endl;
    }
}

void hapusSeluruhData() {
    jumlahBarang = 0;
    cout << "Seluruh data barang berhasil dihapus." << endl;
}

void cariData() {
    int posisi;
    cout << "Masukkan posisi data yang ingin dicari: ";

```

```

    cin >> posisi;
    if (posisi >= 0 && posisi < jumlahBarang) {
        cout << "Data pada posisi " << posisi << ":" << endl;
        cout << "Kode: " << daftarBarang[posisi].kode << ", Nama: " <<
daftarBarang[posisi].nama << ", Harga: " << daftarBarang[posisi].harga
<< ", Jumlah: " << daftarBarang[posisi].jumlah << endl;
    } else {
        cout << "Posisi data tidak valid." << endl;
    }
}

void tambahDataUrutanTertentu() {
    int posisi;
    if (jumlahBarang < MAX_BARANG) {
        cout << "Masukkan posisi data yang ingin ditambahkan: ";
        cin >> posisi;
        if (posisi >= 0 && posisi <= jumlahBarang) {
            for (int i = jumlahBarang - 1; i >= posisi; i--) {
                daftarBarang[i + 1] = daftarBarang[i];
            }
            cout << "Masukkan kode barang: ";
            cin >> daftarBarang[posisi].kode;
            cout << "Masukkan nama barang: ";
            cin >> daftarBarang[posisi].nama;
            cout << "Masukkan harga barang: ";
            cin >> daftarBarang[posisi].harga;
            cout << "Masukkan jumlah barang: ";
            cin >> daftarBarang[posisi].jumlah;
            jumlahBarang++;
            cout << "Data barang berhasil ditambahkan pada posisi " <<
posisi << endl;
        } else {
            cout << "Posisi data tidak valid." << endl;
        }
    } else {
        cout << "Maaf, kapasitas maksimum sudah tercapai." << endl;
    }
}

void tampilkanData() {
    cout << "Daftar Barang:" << endl;
    for (int i = 0; i < jumlahBarang; i++) {
        cout << "Kode: " << daftarBarang[i].kode << ", Nama: " <<
daftarBarang[i].nama << ", Harga: " << daftarBarang[i].harga << ",
Jumlah: " << daftarBarang[i].jumlah << endl;
    }
}

```

```
void menu() {
    cout << "Toko Alat Tulis Barokah" << endl;
    cout << "1. Tambah Data" << endl;
    cout << "2. Hapus Data" << endl;
    cout << "3. Update Data" << endl;
    cout << "4. Hapus Data Urutan Tertentu" << endl;
    cout << "5. Hapus Seluruh Data" << endl;
    cout << "6. Cari data di posisi tertentu" << endl;
    cout << "7. Tampilkan Data" << endl;
    cout << "8. Tambah Data Urutan Tertentu" << endl;
    cout << "9. Exit" << endl;
}

int main() {
    int pilihan;
    do {
        menu();
        cout << "Masukkan pilihan: ";
        cin >> pilihan;
        switch (pilihan) {
            case 1:
                tambahData();
                break;
            case 2:
                hapusData();
                break;
            case 3:
                updateData();
                break;
            case 4:
                hapusDataUrutanTertentu();
                break;
            case 5:
                hapusSeluruhData();
                break;
            case 6:
                cariData();
                break;
            case 7:
                tampilkanData();
                break;
            case 8:
                tambahDataUrutanTertentu();
                break;
            case 9:
```

```
        cout << "Terima kasih telah menggunakan program kasir."
<< endl;
        break;
default:
        cout << "Pilihan tidak valid. Silakan coba lagi." <<
endl;
    }
} while (pilihan != 9);

return 0;
}
```

Output

```

PS C:\Users\LENOVO> & 'c:\Users\LENOVO\vscode\src\cppdbg\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-gtshgulf.hwf' '--stdout=Microsoft-MIEngine-Out-allnoso.zoh' '--stderr=Microsoft-MIEngine-Error-digrn3u2z.lxu' '--pid=Microsoft-MIEngine-Pid-bmoakt10.3sc' '--dbgExe=C:\WinDbg\bin\gdb.exe' '--interpreter=m1'
Toko Alat Tulis Barokah
1. Tambah Data
2. Hapus Data
3. Update Data
4. Hapus Data Urutan Tertentu
5. Hapus Seluruh Data
6. Cari data di posisi tertentu
7. Tampilkan Data
8. Tambah Data Urutan Tertentu
9. Exit
Masukkan pilihan: 1
Masukkan kode barang: A1
Masukkan nama barang: Buku
Masukkan harga barang: 1000
Masukkan jumlah barang: 10
Data barang berhasil ditambahkan.
Toko Alat Tulis Barokah
1. Tambah Data
2. Hapus Data
3. Update Data
4. Hapus Data Urutan Tertentu
5. Hapus Seluruh Data
6. Cari data di posisi tertentu
7. Tampilkan Data
8. Tambah Data Urutan Tertentu
9. Exit
Masukkan pilihan: 1
Masukkan kode barang: A2
Masukkan nama barang: Pensil
Masukkan harga barang: 1000
Masukkan jumlah barang: 10
Data barang berhasil ditambahkan.
Toko Alat Tulis Barokah
1. Tambah Data
2. Hapus Data
3. Update Data
4. Hapus Data Urutan Tertentu
5. Hapus Seluruh Data
6. Cari data di posisi tertentu
7. Tampilkan Data
8. Tambah Data Urutan Tertentu
9. Exit
Masukkan pilihan: 1
Masukkan kode barang: A3
Masukkan nama barang: Penghapus
Masukkan harga barang: 2000
Masukkan jumlah barang: 10
Data barang berhasil ditambahkan.
Toko Alat Tulis Barokah
1. Tambah Data
2. Hapus Data
3. Update Data
4. Hapus Data Urutan Tertentu
5. Hapus Seluruh Data
6. Cari data di posisi tertentu
7. Tampilkan Data
8. Tambah Data Urutan Tertentu
9. Exit
Masukkan pilihan: 2
Masukkan kode barang yang ingin dihapus: A1
Data barang berhasil dihapus.
Toko Alat Tulis Barokah
1. Tambah Data
2. Hapus Data
3. Update Data
4. Hapus Data Urutan Tertentu
5. Hapus Seluruh Data
6. Cari data di posisi tertentu
7. Tampilkan Data
8. Tambah Data Urutan Tertentu
9. Exit

```

```
Daftar Barang:  
Kode: A2, Nama: Pensil, Harga: 1000, Jumlah: 10  
Kode: A3, Nama: Penghapus, Harga: 2000, Jumlah: 10  
Toko Alat Tulis Barokah  
1. Tambah Data  
2. Hapus Data  
3. Update Data  
4. Hapus Data Urutan Tertentu  
5. Hapus Seluruh Data  
6. Cari data di posisi tertentu  
7. Tampilkan Data  
8. Tambah Data Urutan Tertentu  
9. Exit  
Masukkan pilihan: 3  
Masukkan kode barang yang ingin diupdate: A2  
Masukkan harga baru: 5000  
Masukkan jumlah baru: 20  
Data barang berhasil diupdate.  
Toko Alat Tulis Barokah  
1. Tambah Data  
2. Hapus Data  
3. Update Data  
4. Hapus Data Urutan Tertentu  
5. Hapus Seluruh Data  
6. Cari data di posisi tertentu  
7. Tampilkan Data  
8. Tambah Data Urutan Tertentu  
9. Exit  
Masukkan pilihan: 7  
Daftar Barang:  
Kode: A2, Nama: Pensil, Harga: 5000, Jumlah: 20  
Kode: A3, Nama: Penghapus, Harga: 2000, Jumlah: 10  
Toko Alat Tulis Barokah  
1. Tambah Data  
2. Hapus Data  
3. Update Data  
4. Hapus Data Urutan Tertentu
```

```
2. Hapus Data  
3. Update Data  
4. Hapus Data Urutan Tertentu  
5. Hapus Seluruh Data  
6. Cari data di posisi tertentu  
7. Tampilkan Data  
8. Tambah Data Urutan Tertentu  
9. Exit  
Masukkan pilihan: 5  
Seluruh data barang berhasil dihapus.  
Toko Alat Tulis Barokah  
1. Tambah Data  
2. Hapus Data  
3. Update Data  
4. Hapus Data Urutan Tertentu  
5. Hapus Seluruh Data  
6. Cari data di posisi tertentu  
7. Tampilkan Data  
8. Tambah Data Urutan Tertentu  
9. Exit  
Masukkan pilihan: 7  
Daftar Barang:  
Toko Alat Tulis Barokah  
1. Tambah Data  
2. Hapus Data  
3. Update Data  
4. Hapus Data Urutan Tertentu  
5. Hapus Seluruh Data  
6. Cari data di posisi tertentu  
7. Tampilkan Data  
8. Tambah Data Urutan Tertentu  
9. Exit  
Masukkan pilihan: 9  
Terima kasih telah menggunakan program kasir.
```

Penjelasan

Membuat program kasir menggunakan array, hash table, stack, dan linked list, diimplementasikan dengan bahasa c++. Dengan menginput data barang dan menghapus beberapa barang.

Menggunakan modul 3, 4, 5 dan 6