

Nama : Jenderal Nicolas Tampubolon


Kelas : 22IF05

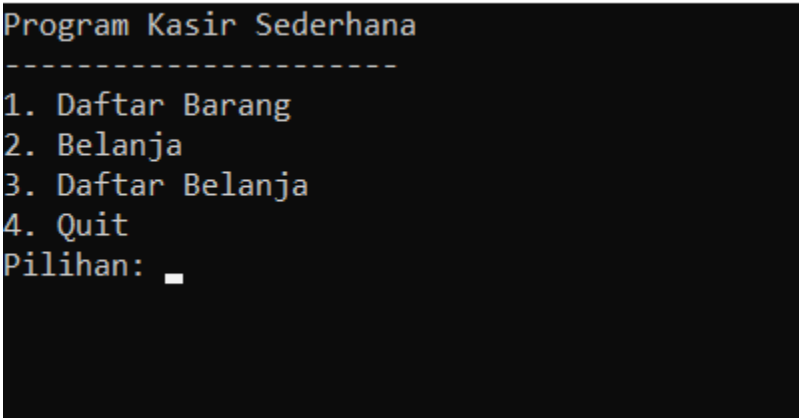
NIM : 22.11.4856

Tugas praktikum pengganti

Perintah : Buatlah sebuah program menggunakan Linked List, Searching, Dan Queue Untuk Membuat sebuah program kasir sederhana yang dapat menginputkan Sebuah barang dan ketika sudah selesai diinputkan akan tampil barang dan jumlah keseluruhan harganya

1. Menu utama

 D:\AMIKOM\Semester 2\Struktur Data\13\tes.exe



```
Program Kasir Sederhana
-----
1. Daftar Barang
2. Belanja
3. Daftar Belanja
4. Quit
Pilihan: _
```

Menu ini merupakan menu utama dari program yg saya buat, disini kita dapat memilih menggunakan angka 1-4 sesuai yg diinginkan. Untuk pilihan 1 program akan menampilkan daftar barang beserta harga dari masing-masing barangnya.

Kemudian menu 2 program akan menampilkan pilihan barang untuk kemudian kita pilih barang dan inputkan jumlah barang yg ingin dibelanjakan. Setelah itu akan ada perulangan untuk lanjut belanja atau kembali ke menu utama. Setelah inputkan belanjaan maka data akan disimpan.

Pada menu 3 program akan menampilkan daftar nama dan jumlah masing-masing barang yg dibeli, serta menampilkan total keseluruhan harga berdasarkan semua barang yg dibeli.

Menu 4 merupakan opsi untuk keluar/mengakhiri program.

2. Menu daftar barang

D:\AMIKOM\Semester 2\Struktur Data\13\tes.exe

```
Program Kasir Sederhana
-----
1. Daftar Barang
2. Belanja
3. Daftar Belanja
4. Quit
Pilihan: 1
Daftar Barang:
-----
1. Nama Barang: Sapu
   Harga: Rp 15000
2. Nama Barang: Gergaji
   Harga: Rp 26000
3. Nama Barang: Palu
   Harga: Rp 30000
4. Nama Barang: Gayung
   Harga: Rp 12000
5. Nama Barang: Kemoceng
   Harga: Rp 18000
6. Nama Barang: Obat Nyamuk
   Harga: Rp 20000
7. Nama Barang: Sabun Mandi
   Harga: Rp 12500

Program Kasir Sederhana
-----
1. Daftar Barang
2. Belanja
3. Daftar Belanja
4. Quit
Pilihan: _
```

3. Menu belanja

```
Program Kasir Sederhana
-----
1. Daftar Barang
2. Belanja
3. Daftar Belanja
4. Quit
Pilihan: 2
Daftar Barang:
-----
1. Nama Barang: Sapu
   Harga: Rp 15000
2. Nama Barang: Gergaji
   Harga: Rp 26000
3. Nama Barang: Palu
   Harga: Rp 30000
4. Nama Barang: Gayung
   Harga: Rp 12000
5. Nama Barang: Kemoceng
   Harga: Rp 18000
6. Nama Barang: Obat Nyamuk
   Harga: Rp 20000
7. Nama Barang: Sabun Mandi
   Harga: Rp 12500
Masukkan nomor barang yang akan dibeli: 2
Masukkan jumlah barang: 1
Barang berhasil ditambahkan ke keranjang.
Total Harga: Rp 26000
Lanjut belanja? (y/n): y
Masukkan nomor barang yang akan dibeli: 7
Masukkan jumlah barang: 3
Barang berhasil ditambahkan ke keranjang.
Total Harga: Rp 37500
Lanjut belanja? (y/n): n
```

4. Menu daftar belanja

```
Program Kasir Sederhana
-----
1. Daftar Barang
2. Belanja
3. Daftar Belanja
4. Quit
Pilihan: 3
List Belanja:
-----
Nama Barang: Gergaji(x1), Sabun Mandi(x3)
Total Harga: Rp 63500
```

5. Menu Quit

```
Program Kasir Sederhana
-----
1. Daftar Barang
2. Belanja
3. Daftar Belanja
4. Quit
Pilihan: 4

-----
Process exited after 700.8 seconds with return value 0
Press any key to continue . . .
```

6. Kode program

```
#include <iostream>
#include <string>
#include <sstream>
using namespace std;

struct Barang {
    string nama;
    int harga;
    Barang* next;
};

Barang* createBarang(string nama, int harga) {
    Barang* newBarang = new Barang;
    newBarang->nama = nama;
    newBarang->harga = harga;
    newBarang->next = NULL;
```

```
    return newBarang;
}
```

```
void tambahBarang(Barang** head, string nama, int harga) {
    Barang* newBarang = createBarang(nama, harga);
    if (*head == NULL) {
        *head = newBarang;
    } else {
        Barang* cur = *head;
        while (cur->next) {
            cur = cur->next;
        }
        cur->next = newBarang;
    }
}
```

```
void tampilkanBarang(Barang* head) {
    Barang* cur = head;
    int nomor = 1; // Penomoran barang
    cout << "Daftar Barang:" << endl;
    cout << "-----" << endl;
    while (cur) {
        cout << nomor << ". Nama Barang: " << cur->nama << endl;
        cout << "  Harga: Rp " << cur->harga << endl;
        cur = cur->next;
        nomor++;
    }
}
```

```
Barang* getBarangByNomor(Barang* head, int nomor) {
    Barang* cur = head;
    int i = 1;
    while (cur && i != nomor) {
        cur = cur->next;
        i++;
    }
    return cur;
}
```

```
struct QueueNode {
    int nomor;
    int jumlah;
    QueueNode* next;
```

```
};
```

```
QueueNode* createQueueNode(int nomor, int jumlah) {  
    QueueNode* newQueueNode = new QueueNode;  
    newQueueNode->nomor = nomor;  
    newQueueNode->jumlah = jumlah;  
    newQueueNode->next = NULL;  
    return newQueueNode;  
}
```

```
struct Queue {  
    QueueNode* front;  
    QueueNode* rear;  
};
```

```
void initQueue(Queue& q) {  
    q.front = NULL;  
    q.rear = NULL;  
}
```

```
bool isEmpty(Queue q) {  
    return (q.front == NULL && q.rear == NULL);  
}
```

```
void enqueue(Queue& q, int nomor, int jumlah) {  
    QueueNode* newQueueNode = createQueueNode(nomor, jumlah);  
    if (isEmpty(q)) {  
        q.front = newQueueNode;  
        q.rear = newQueueNode;  
    } else {  
        q.rear->next = newQueueNode;  
        q.rear = newQueueNode;  
    }  
}
```

```
void displayQueue(Queue q, Barang* daftarBarang) {  
    QueueNode* cur = q.front;  
    cout << "List Belanja:" << endl;  
    cout << "-----" << endl;  
    int totalHarga = 0; // Total harga keseluruhan  
    stringstream listBarang;  
    while (cur) {  
        Barang* barang = getBarangByNomor(daftarBarang, cur->nomor);
```

```

        if (barang != NULL) {
            listBarang << barang->nama << "(x" << cur->jumlah << ")", ";
            totalHarga += barang->harga * cur->jumlah;
        }
        cur = cur->next;
    }
    string listBarangStr = listBarang.str();
    if (!listBarangStr.empty()) {
        // Menghilangkan koma dan spasi terakhir
        listBarangStr = listBarangStr.substr(0, listBarangStr.length() - 2);
    }
    cout << "Nama Barang: " << listBarangStr << endl;
    cout << "Total Harga: Rp " << totalHarga << endl;
}

```

```

int main() {
    Barang* daftarBarang = NULL;

    tambahBarang(&daftarBarang, "Sapu", 15000);
    tambahBarang(&daftarBarang, "Gergaji", 26000);
    tambahBarang(&daftarBarang, "Palu", 30000);
    tambahBarang(&daftarBarang, "Gayung", 12000);
    tambahBarang(&daftarBarang, "Kemoceng", 18000);
    tambahBarang(&daftarBarang, "Obat Nyamuk", 20000);
    tambahBarang(&daftarBarang, "Sabun Mandi", 12500);
}

```

```

Queue keranjang;
initQueue(keranjang);

```

```

int pilihan, jumlah;
char lanjut;

```

```

while (true) {
    cout << "Program Kasir Sederhana" << endl;
    cout << "-----" << endl;
    cout << "1. Daftar Barang" << endl;
    cout << "2. Belanja" << endl;
    cout << "3. Daftar Belanja" << endl;
    cout << "4. Quit" << endl;
    cout << "Pilihan: ";
    cin >> pilihan;

    switch (pilihan) {

```

```

case 1:
    tampilkanBarang(daftarBarang);
    break;
case 2:
    tampilkanBarang(daftarBarang);
    do {
        cout << "Masukkan nomor barang yang akan dibeli: ";
        cin >> pilihan;
        Barang* barang = getBarangByNomor(daftarBarang, pilihan);
        if (barang != NULL) {
            cout << "Masukkan jumlah barang: ";
            cin >> jumlah;
            enqueue(keranjang, pilihan, jumlah);
            cout << "Barang berhasil ditambahkan ke keranjang." << endl;
            cout << "Total Harga: Rp " << (barang->harga * jumlah) << endl;
        } else {
            cout << "Barang tidak ditemukan." << endl;
        }
        cout << "Lanjut belanja? (y/n): ";
        cin >> lanjut;
    } while (lanjut == 'y' || lanjut == 'Y');
    break;
case 3:
    displayQueue(keranjang, daftarBarang);
    break;
case 4:
    exit(0);
default:
    cout << "Pilihan tidak valid." << endl;
}

cout << endl;
}

return 0;
}

```