

Studi Perancangan dan Implementasi Sistem Kasir Sederhana Menggunakan Bahasa Pemrograman C++ pada Platform Code::Blocks

Hafizh Naufal Azzam¹, Arya Sabily Ubaidillah²

^{1,2}Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Kota Semarang

Artikel Info

Kata kunci:

C++
program kasir
dasar pemrograman
sistem transaksi
percabangan dan perulangan

ABSTRAK

Perkembangan teknologi informasi telah mendorong pemanfaatan sistem terkomputerisasi dalam berbagai aktivitas, termasuk dalam proses transaksi jual beli. Sistem kasir berbasis komputer menjadi salah satu solusi untuk meningkatkan efisiensi, keakuratan, dan kecepatan dalam melakukan perhitungan transaksi. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah program kasir sederhana menggunakan bahasa pemrograman C++ sebagai penerapan konsep dasar pemrograman. Program yang dikembangkan mampu menampilkan daftar barang beserta harga, menerima input pilihan barang dan jumlah pembelian, menghitung total harga, diskon, subtotal, serta menjumlahkan keseluruhan transaksi dalam satu kali proses penggunaan. Metode yang digunakan dalam pembuatan program meliputi penerapan variabel dan tipe data, penggunaan array untuk pengelolaan data barang, struktur percabangan untuk penentuan diskon berdasarkan kondisi tertentu, serta struktur perulangan untuk mengatur proses transaksi berulang. Pengujian dilakukan dengan menjalankan program dan memasukkan berbagai variasi input untuk memastikan seluruh fungsi berjalan sesuai dengan perancangan. Hasil pengujian menunjukkan bahwa program dapat berfungsi dengan baik dan menghasilkan perhitungan yang akurat sesuai dengan ketentuan yang telah ditetapkan. Berdasarkan hasil implementasi dan pengujian, dapat disimpulkan bahwa program kasir sederhana berbasis C++ ini mampu membantu proses transaksi penjualan secara lebih efektif dan efisien dibandingkan perhitungan manual. Selain itu, program ini juga dapat dijadikan sebagai media pembelajaran dalam memahami dan mengaplikasikan konsep dasar pemrograman pada permasalahan nyata, khususnya dalam pengembangan aplikasi transaksi sederhana.

Penulis Korespondensi :

Hafizh Naufal Azzam,
Program Studi Teknik Informatika
Fakultas Ilmu Komputer
Universitas Dian Nuswantoro, Kota Semarang, 50131
Email: 111202516162@mhs.dinus.ac.id

1. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi yang semakin pesat telah memberikan dampak yang signifikan terhadap berbagai sektor kehidupan, termasuk sektor perdagangan dan jasa [1]. Pemanfaatan teknologi komputer dalam kegiatan jual beli menjadi salah satu solusi untuk meningkatkan efisiensi, akurasi, dan kecepatan dalam proses transaksi [2]. Sistem penjualan yang masih dilakukan secara manual memiliki

berbagai keterbatasan, seperti tingginya risiko kesalahan perhitungan, pencatatan data yang kurang rapi, serta waktu pelayanan yang relatif lebih lama [3]. Oleh karena itu, penggunaan sistem terkomputerisasi menjadi kebutuhan penting dalam mendukung kegiatan operasional usaha

Dalam bidang pendidikan, khususnya pada program studi yang berkaitan dengan teknik informatika, penguasaan keterampilan pemrograman merupakan salah satu kompetensi dasar yang harus dimiliki oleh mahasiswa [4]. Mata kuliah Dasar Pemrograman dirancang untuk membekali mahasiswa dengan pemahaman konsep-konsep fundamental pemrograman komputer, mulai dari pengenalan algoritma, penggunaan variabel dan tipe data, hingga penerapan struktur kontrol seperti percabangan dan perulangan [5]. Penguasaan konsep-konsep tersebut sangat penting karena menjadi landasan dalam pengembangan aplikasi yang lebih kompleks pada tahap pembelajaran selanjutnya.

Bahasa pemrograman C++ dipilih sebagai bahasa pemrograman yang digunakan dalam tugas ini karena memiliki karakteristik yang kuat, terstruktur, serta banyak digunakan dalam dunia pendidikan maupun industri [6], [7]. C++ memungkinkan mahasiswa untuk memahami alur logika program secara jelas dan sistematis, serta melatih kemampuan berpikir analitis dalam menyusun algoritma pemecahan masalah. Selain itu, penggunaan *Integrated Development Environment* (IDE) Code::Blocks memudahkan proses penulisan, pengujian, dan *debugging* program sehingga mahasiswa dapat lebih fokus pada pemahaman konsep pemrograman itu sendiri [8].

Berdasarkan latar belakang tersebut, pada tugas kelompok ini penulis merancang dan mengimplementasikan sebuah program kasir sederhana yang mensimulasikan sistem penjualan pada sebuah toko pakaian. Program ini dirancang untuk menampilkan daftar barang beserta harga yang telah ditentukan, menerima input dari pengguna berupa pilihan barang dan jumlah pembelian, serta melakukan proses perhitungan total harga secara otomatis. Selain itu, program juga dilengkapi dengan fitur perhitungan diskon berdasarkan ketentuan tertentu, sehingga total pembayaran yang dihasilkan lebih akurat dan realistis.

Pembuatan program kasir sederhana ini diharapkan dapat membantu mahasiswa dalam memahami penerapan konsep dasar pemrograman secara langsung dalam studi kasus yang dekat dengan kehidupan sehari-hari. Dengan mengimplementasikan program ini, mahasiswa tidak hanya mempelajari teori pemrograman, tetapi juga belajar bagaimana mengubah permasalahan nyata menjadi sebuah solusi berbasis komputer. Selain itu, penyusunan laporan ini bertujuan untuk mendokumentasikan proses perancangan, implementasi, serta hasil pengujian program secara sistematis dan terstruktur, sehingga dapat menjadi bahan evaluasi dan referensi pembelajaran di masa mendatang.

Tujuan utama dari pembuatan program kasir sederhana ini adalah untuk membantu mahasiswa memahami dan menguasai konsep dasar pemrograman komputer menggunakan bahasa pemrograman C++. Melalui proses perancangan dan implementasi program, mahasiswa diharapkan mampu memahami alur logika pemrograman secara runtut serta mengenal struktur dasar penulisan kode yang baik dan benar sesuai dengan kaidah bahasa C++.

Selain itu, pembuatan program ini bertujuan untuk menerapkan berbagai konsep dasar pemrograman yang telah dipelajari, seperti penggunaan variabel, *array*, struktur percabangan, dan perulangan. Penerapan konsep-konsep tersebut dalam sebuah program nyata diharapkan dapat meningkatkan pemahaman mahasiswa secara praktis, sehingga tidak hanya terbatas pada teori yang diperoleh di dalam kelas.

Tujuan berikutnya adalah untuk merancang dan mengimplementasikan sebuah program transaksi penjualan sederhana yang menyerupai sistem kasir pada toko pakaian. Dengan adanya program ini, mahasiswa dapat memahami bagaimana proses transaksi penjualan dilakukan secara terkomputerisasi, mulai dari pemilihan barang, perhitungan jumlah pembelian, hingga penentuan total pembayaran yang harus dibayarkan oleh pembeli.

Melalui tugas ini, mahasiswa juga diharapkan dapat melatih kemampuan berpikir logis, analitis, dan sistematis dalam menyelesaikan permasalahan pemrograman. Proses penyusunan algoritma dan pemecahan masalah yang muncul selama pembuatan program akan membantu mahasiswa dalam mengembangkan pola pikir yang terstruktur dan kritis.

Selain aspek teknis, pembuatan program dan laporan ini bertujuan untuk meningkatkan kemampuan kerja sama, komunikasi, dan tanggung jawab antar anggota kelompok. Dengan bekerja secara tim, mahasiswa dilatih untuk saling berkoordinasi, berbagi tugas, serta menyelesaikan permasalahan bersama demi mencapai hasil yang optimal.

2. METODE

Bab ini membahas metodologi yang digunakan dalam perancangan dan pembuatan program kasir sederhana berbasis bahasa pemrograman C++. Metodologi disusun untuk menjelaskan konsep-konsep dasar pemrograman yang diterapkan, teknik pemrograman yang digunakan, serta alur implementasi program secara

sistematis. Pembahasan pada bab ini bertujuan untuk memberikan gambaran yang jelas mengenai dasar teoritis dan teknis yang mendasari pembuatan program, sehingga pembaca dapat memahami proses pengembangan program secara menyeluruh.

Adapun pembahasan dalam bab ini mencakup pengertian dasar pemrograman, penggunaan bahasa pemrograman C++ dan lingkungan pengembangannya, pemanfaatan variabel dan tipe data, penerapan struktur percabangan dan perulangan, serta penjelasan alur dan implementasi kode program kasir sederhana. Setiap subbab disusun secara bertahap untuk menunjukkan keterkaitan antara konsep pemrograman dengan implementasi program yang telah dibuat.

2.1 Pengertian Dasar Pemrograman

Pemrograman merupakan proses sistematis yang dilakukan untuk menyusun sekumpulan instruksi agar komputer dapat menjalankan tugas tertentu sesuai dengan tujuan yang diinginkan [9]. Proses ini melibatkan beberapa tahapan, mulai dari perancangan algoritma, penulisan kode program, pengujian, hingga pemeliharaan program [10]. Instruksi tersebut ditulis menggunakan bahasa pemrograman yang memiliki aturan sintaksis dan struktur tertentu sehingga dapat dipahami oleh komputer [11].

Dalam konteks pengembangan perangkat lunak, pemrograman tidak hanya berfokus pada penulisan kode, tetapi juga pada bagaimana menyusun logika yang benar dan efisien untuk menyelesaikan suatu permasalahan [12], [13]. Kemampuan berpikir logis dan sistematis sangat diperlukan agar program yang dihasilkan dapat berjalan dengan baik dan menghasilkan keluaran yang sesuai dengan kebutuhan pengguna.

Pada pembuatan program kasir sederhana ini, konsep dasar pemrograman diterapkan untuk mensimulasikan proses transaksi penjualan pada sebuah toko pakaian. Program dirancang agar mampu menerima *input* dari pengguna, melakukan perhitungan otomatis, serta menampilkan hasil transaksi secara jelas dan terstruktur. Dengan demikian, mahasiswa dapat memahami bagaimana konsep dasar pemrograman diterapkan dalam permasalahan nyata.

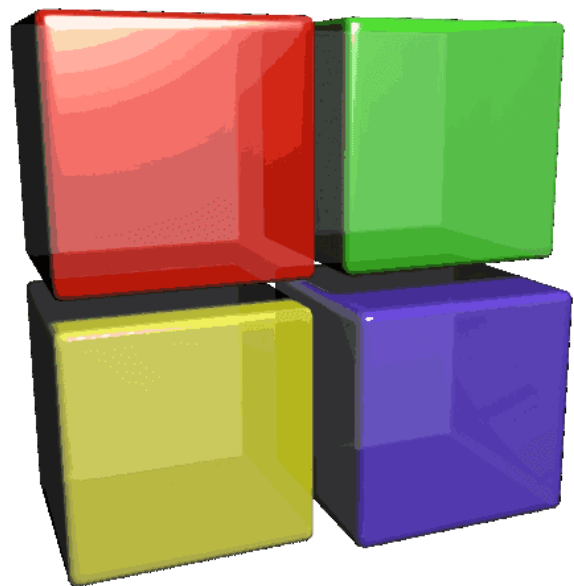
2.2 Bahasa Pemrograman C++ dan Lingkungan Pengembangan

Bahasa pemrograman C++ merupakan bahasa pemrograman tingkat menengah yang banyak digunakan dalam dunia pendidikan dan industri [14]. Bahasa ini memiliki struktur yang jelas serta mendukung pemrograman terstruktur dan modular [15]. C++ dipilih dalam pembuatan program ini karena mampu melatih mahasiswa dalam memahami alur logika program serta pengelolaan data secara sistematis [16].

Dalam implementasi program kasir sederhana, C++ digunakan untuk mengatur alur transaksi, melakukan perhitungan harga dan diskon, serta mengelola *input* dan *output* program. Penggunaan bahasa ini membantu mahasiswa memahami bagaimana instruksi pemrograman diterjemahkan menjadi proses yang dapat dijalankan oleh komputer.



(a) Bahasa Pemrograman C++



(b) IDE Code::Blocks

Gambar 1 Lingkungan Pengembangan Program

Lingkungan pengembangan yang digunakan adalah Code::Blocks, yaitu *Integrated Development Environment* (IDE) yang mendukung bahasa pemrograman C++. Code::Blocks menyediakan fasilitas penulisan kode, kompilasi, dan *debugging* sehingga memudahkan proses pengembangan program. Dengan adanya IDE ini, kesalahan sintaksis dan logika dapat dideteksi lebih cepat, sehingga proses pembelajaran menjadi lebih efektif.

2.3 Variabel dan Tipe Data

Variabel merupakan elemen dasar dalam pemrograman yang berfungsi sebagai tempat penyimpanan data sementara selama program dijalankan [17]. Setiap variabel memiliki nama, tipe data, dan nilai yang dapat berubah sesuai dengan proses yang terjadi di dalam program. Tipe data digunakan untuk menentukan jenis nilai yang dapat disimpan oleh suatu variabel [18].

Tabel 1 Contoh Variabel dan Tipe Data Pada C++

main.cpp
<pre>#include <iostream> using namespace std; int main() { // Character variable char c = 'A'; cout << c; return 0; }</pre>

Dalam program kasir sederhana ini, digunakan berbagai tipe data seperti *int* untuk menyimpan nilai bilangan bulat (jumlah barang, harga, pilihan menu), *double* untuk menyimpan nilai pecahan (diskon, subtotal, total keseluruhan), *char* untuk menyimpan pilihan lanjut belanja, serta *string* untuk menyimpan nama barang. Selain itu, *array* digunakan untuk menyimpan daftar nama barang dan harga agar data lebih terstruktur dan mudah diakses. Penggunaan variabel dan tipe data yang tepat memungkinkan program melakukan perhitungan transaksi secara otomatis dan akurat. Hal ini membantu mahasiswa memahami bagaimana data disimpan, diolah, dan ditampilkan dalam sebuah program komputer.

2.4 Percabangan (If-Else)

Percabangan merupakan struktur kontrol yang digunakan untuk menentukan alur eksekusi program berdasarkan kondisi tertentu [19]. Dalam bahasa pemrograman C++, percabangan umumnya menggunakan pernyataan *if*, *if-else*, dan *else if* [19]. Struktur ini memungkinkan program untuk mengambil keputusan sesuai dengan *input* yang diberikan oleh pengguna.

Tabel 2 Contoh Percabangan Pada C++

main.cpp
<pre>int number = 10; if (number > 5) { std::cout << "The number is greater than 5." << std::endl; }</pre>

Pada program kasir sederhana ini, percabangan digunakan untuk memvalidasi pilihan barang yang dimasukkan oleh pengguna. Jika pilihan berada dalam rentang yang tersedia, program akan melanjutkan proses transaksi. Sebaliknya, jika pilihan tidak valid, program akan menampilkan pesan kesalahan. Selain itu, percabangan juga digunakan untuk menentukan besarnya diskon berdasarkan jenis barang dan jumlah pembelian. Penerapan percabangan ini membuat program mampu menyesuaikan proses perhitungan sesuai dengan kondisi yang terjadi. Dengan demikian, program dapat berjalan secara dinamis dan memberikan hasil yang sesuai dengan ketentuan yang telah ditetapkan.

2.5 Perulangan (Looping)

Perulangan merupakan struktur kontrol yang digunakan untuk menjalankan suatu proses secara berulang selama kondisi tertentu terpenuhi [20]. Dalam bahasa pemrograman C++, dikenal beberapa jenis perulangan seperti *for*, *while*, dan *do-while* [21]. Perulangan digunakan untuk menghindari penulisan kode yang berulang-ulang dan membuat program menjadi lebih efisien [21].

Tabel 3 Contoh Perulangan Pada C++

main.cpp
<pre>#include <iostream> using namespace std; int main() { for (int i = 0; i < 5; i++) { // Loop from i = 0 to 4 cout << "Iteration " << i << endl; } return 0; }</pre>

Pada program kasir sederhana ini, perulangan *for* digunakan untuk menampilkan daftar barang dan harga. Sementara itu, perulangan *do-while* digunakan untuk memungkinkan pengguna melakukan transaksi lebih dari satu kali tanpa harus menjalankan ulang program. Program akan terus berjalan selama pengguna memilih untuk melanjutkan proses belanja. Penggunaan perulangan ini membuat program menjadi lebih interaktif dan sesuai dengan kebutuhan sistem kasir, di mana pengguna dapat melakukan beberapa transaksi dalam satu kali penggunaan program.

2.6 Penjelasan Alur dan Implementasi Program

Program kasir sederhana diawali dengan pemanggilan fungsi *header()* yang berfungsi untuk menampilkan identitas toko dalam bentuk teks. Hal ini bertujuan untuk memberikan tampilan awal yang rapi dan informatif kepada pengguna. Setelah itu, program mendeklarasikan berbagai variabel yang digunakan untuk menyimpan data transaksi, seperti jumlah barang, harga, diskon, subtotal, dan total keseluruhan. Selanjutnya, program menampilkan daftar barang beserta harga menggunakan *array* dan perulangan. Pengguna diminta untuk memilih barang dan memasukkan jumlah pembelian. Program kemudian menghitung total harga berdasarkan harga barang dan jumlah pembelian yang dimasukkan. Proses perhitungan diskon dilakukan menggunakan struktur percabangan sesuai dengan ketentuan yang telah ditetapkan. Setelah diskon dihitung, program menentukan subtotal yang harus dibayarkan dan menambahkan nilai tersebut ke total keseluruhan belanja. Pengguna kemudian diberikan pilihan untuk melanjutkan belanja atau mengakhiri transaksi. Jika pengguna memilih untuk melanjutkan, program akan mengulangi proses transaksi. Jika tidak, program akan menampilkan total keseluruhan belanja dan memanggil fungsi *footer()* sebagai penutup program. Dengan adanya pemisahan fungsi seperti *header()* dan *footer()*, program menjadi lebih terstruktur dan mudah dipahami. Implementasi ini juga membantu mahasiswa memahami konsep modularitas dalam pemrograman, di mana program dibagi menjadi bagian-bagian kecil yang memiliki fungsi masing-masing.

3. PEMBAHASAN HASIL

Bab ini membahas hasil perancangan dan implementasi program kasir sederhana berbasis bahasa pemrograman C++ yang telah dibuat. Pembahasan difokuskan pada gambaran umum program, struktur program, serta penjelasan kode dan mekanisme kerja program dalam menjalankan proses transaksi penjualan. Melalui pembahasan ini, diharapkan pembaca dapat memahami bagaimana konsep dasar pemrograman yang telah dijelaskan pada bab sebelumnya diterapkan secara nyata dalam sebuah program yang berfungsi dengan baik.

3.1 Gambaran Umum Program

Program yang dikembangkan merupakan program kasir sederhana yang digunakan untuk mensimulasikan proses transaksi penjualan pada Toko Baju Mulyo. Program ini dirancang untuk membantu proses perhitungan transaksi secara otomatis, mulai dari pemilihan barang, penentuan jumlah pembelian, hingga perhitungan total harga dan diskon. Dengan adanya program ini, proses transaksi dapat dilakukan

dengan lebih cepat, akurat, dan efisien dibandingkan dengan perhitungan manual. Pada saat program dijalankan, pengguna akan disajikan tampilan awal berupa identitas toko yang ditampilkan melalui fungsi *header*. Selanjutnya, program menampilkan daftar barang yang tersedia beserta harga masing-masing. Pengguna kemudian diminta untuk memilih barang yang akan dibeli dan memasukkan jumlah pembelian sesuai dengan kebutuhan. Program akan memproses *input* tersebut dan menghitung total harga berdasarkan harga satuan dan jumlah barang yang dibeli.

Selain menghitung total harga, program juga dilengkapi dengan fitur perhitungan diskon. Diskon diberikan berdasarkan jenis barang dan jumlah pembelian tertentu yang telah ditentukan sebelumnya. Setelah diskon dihitung, program akan menampilkan subtotal yang harus dibayarkan oleh pengguna. Program ini juga memungkinkan pengguna untuk melakukan pembelian lebih dari satu kali dalam satu kali eksekusi program, sehingga menyerupai sistem kasir pada toko secara nyata.

Di akhir transaksi, program akan menampilkan total keseluruhan belanja yang merupakan akumulasi dari seluruh subtotal pembelian. Tampilan penutup program disajikan melalui fungsi *footer* yang berisi ucapan terima kasih kepada pelanggan. Secara keseluruhan, program kasir sederhana ini telah berhasil memenuhi tujuan utama, yaitu mensimulasikan sistem transaksi penjualan dengan memanfaatkan konsep dasar pemrograman C++.

3.2 Struktur Program

Struktur program kasir sederhana ini disusun secara terorganisir agar mudah dipahami dan dikembangkan. Program terdiri dari beberapa bagian utama yang masing-masing memiliki fungsi dan peran tersendiri dalam menjalankan keseluruhan sistem transaksi. Pembagian struktur program ini juga mencerminkan penerapan konsep pemrograman terstruktur dan modular.

Bagian pertama adalah fungsi *header()*, yang berfungsi untuk menampilkan judul dan identitas toko pada awal program. Fungsi ini bertujuan untuk memberikan tampilan awal yang informatif dan meningkatkan keterbacaan program. Dengan memisahkan tampilan *header* ke dalam fungsi tersendiri, kode program menjadi lebih rapi dan mudah dikelola.

Bagian kedua adalah fungsi *footer()*, yang berfungsi sebagai penutup program. Fungsi ini menampilkan pesan ucapan terima kasih kepada pengguna setelah seluruh proses transaksi selesai. Penggunaan fungsi *footer* membantu memberikan kesan akhir yang lebih profesional dan terstruktur pada program. Bagian utama dari program terletak pada fungsi *main()*. Fungsi ini berperan sebagai inti dari keseluruhan program karena di dalamnya terdapat alur utama proses transaksi. Pada fungsi ini dilakukan deklarasi variabel, penampikan daftar barang, penerimaan *input* dari pengguna, perhitungan harga dan diskon, serta pengendalian perulangan transaksi. Seluruh proses utama program dikendalikan dari fungsi ini, sehingga fungsi *main()* menjadi pusat eksekusi program kasir sederhana.

Tabel 4 Potongan Kode Program

main.cpp
<pre>#include <iostream> using namespace std; int main() { int jml, brg, total, i; double diskon = 0, subtotal, tk = 0; char lanjut; // Array data barang dan harga string pilihan[4] = { "Kaos Polos", "Kaos Polos Lengan Panjang", "Celana Pendek", "Celana Panjang" }; int harga[4] = {15000, 20000, 10000, 18000}; // Menampilkan daftar barang cout << "List Harga Barang\n"; for (i = 0; i < 4; i++) { cout << i + 1 << ". " << pilihan[i] << " : Rp" << harga[i] << endl;</pre>

```

}

do {
    cout << "\nPilih barang (1-4): ";
    cin >> brg;

    if (brg >= 1 && brg <= 4) {
        cout << "Jumlah Barang: ";
        cin >> jml;

        total = harga[brg - 1] * jml;
        diskon = 0;

        // Percabangan diskon
        if (brg == 1 && jml >= 7)
            diskon = total * 0.07;
        else if (brg == 2 && jml >= 10)
            diskon = total * 0.10;
        else if (brg == 3 && jml >= 15)
            diskon = total * 0.05;
        else if (brg == 4 && jml >= 5)
            diskon = total * 0.04;

        subtotal = total - diskon;
        tk += subtotal;

        cout << "Total Harga : Rp" << total << endl;
        cout << "Diskon      : Rp" << diskon << endl;
        cout << "Subtotal   : Rp" << subtotal << endl;
    } else {
        cout << "Pilihan tidak tersedia." << endl;
    }

    do {
        cout << "\nLanjut belanja? (y/n): ";
        cin >> lanjut;
    } while (lanjut != 'y' && lanjut != 'Y' && lanjut != 'n' && lanjut != 'N');

    } while (lanjut == 'y' || lanjut == 'Y');

    cout << "\nTotal Keseluruhan Belanja: Rp" << tk << endl;
    return 0;
}

```

Pembagian program ke dalam beberapa fungsi tersebut menunjukkan penerapan konsep modularitas, di mana program dipecah menjadi bagian-bagian kecil yang memiliki tugas masing-masing. Hal ini tidak hanya memudahkan pemahaman program, tetapi juga mempermudah proses pengembangan dan perawatan kode di masa mendatang.

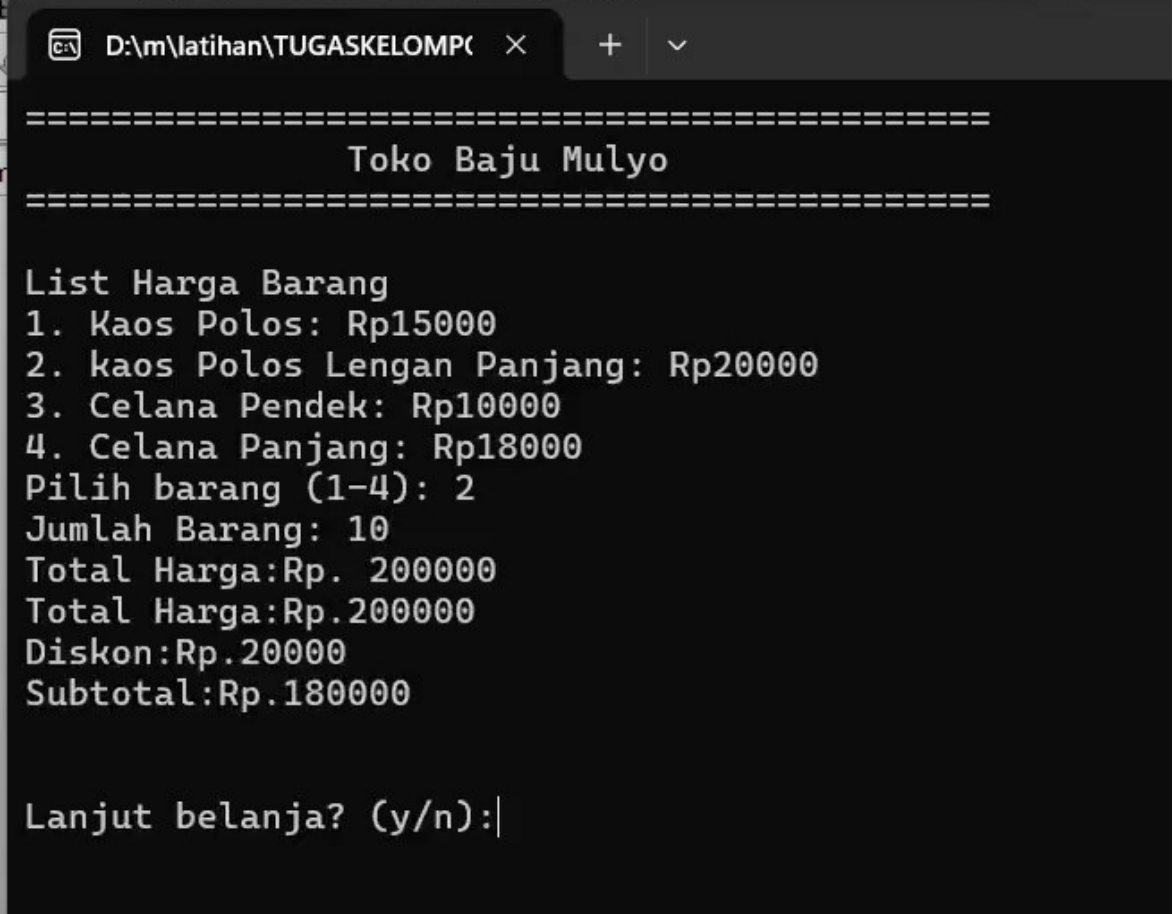
3.3 Penjelasan Kode Program

Kode program kasir sederhana ini menggunakan berbagai elemen dasar pemrograman C++ seperti variabel, *array*, percabangan, dan perulangan. Setiap elemen memiliki peran penting dalam mendukung jalannya program secara keseluruhan.

Array pilihan[] digunakan untuk menyimpan nama-nama barang yang dijual pada Toko Baju Mulyo. Dengan menggunakan *array*, data barang dapat disimpan secara terstruktur dan mudah diakses menggunakan indeks tertentu. Hal ini memudahkan program dalam menampilkan daftar barang kepada pengguna serta mengaitkan nama barang dengan harga yang sesuai.

`Array` `harga[]` digunakan untuk menyimpan harga masing-masing barang. `Array` ini memiliki keterkaitan langsung dengan `array` `pilihan[]`, di mana indeks yang sama menunjukkan pasangan nama barang dan harga. Penggunaan `array` `harga` mempermudah proses perhitungan total harga karena program dapat mengambil nilai harga berdasarkan pilihan barang yang dimasukkan oleh pengguna.

Variabel `jml` digunakan untuk menyimpan jumlah barang yang dibeli oleh pengguna. Nilai variabel ini sangat berpengaruh dalam proses perhitungan total harga dan penentuan diskon. Variabel `total` digunakan untuk menyimpan hasil perhitungan harga sebelum diskon, yang diperoleh dari perkalian harga barang dengan jumlah pembelian.



```
=====
                        Toko Baju Mulyo
=====

List Harga Barang
1. Kaos Polos: Rp15000
2. kaos Polos Lengan Panjang: Rp20000
3. Celana Pendek: Rp10000
4. Celana Panjang: Rp18000
Pilih barang (1-4): 2
Jumlah Barang: 10
Total Harga:Rp. 200000
Total Harga:Rp.200000
Diskon:Rp.20000
Subtotal:Rp.180000

Lanjut belanja? (y/n):|
```

Gambar 2 Hasil Akhir Program

Variabel `diskon` digunakan untuk menyimpan nilai potongan harga yang diberikan kepada pengguna. Besarnya diskon ditentukan menggunakan struktur percabangan berdasarkan jenis barang dan jumlah pembelian. Variabel `subtotal` digunakan untuk menyimpan harga akhir setelah diskon dikurangkan dari total harga. Sementara itu, variabel `tk` digunakan untuk menyimpan total keseluruhan belanja dari seluruh transaksi yang dilakukan selama program berjalan. Struktur percabangan `if-else` digunakan untuk menentukan besarnya diskon yang diberikan. Barang pertama mendapatkan diskon sebesar 7% jika jumlah pembelian minimal tujuh item. Barang kedua mendapatkan diskon 10% jika pembelian mencapai sepuluh item atau lebih. Barang ketiga mendapatkan diskon 5% dengan minimal pembelian lima belas item, sedangkan barang keempat mendapatkan diskon 4% jika dibeli minimal lima item. Penerapan percabangan ini membuat program mampu menyesuaikan perhitungan diskon secara otomatis sesuai dengan ketentuan yang berlaku. Selain itu, program menggunakan perulangan `do-while` untuk memungkinkan pengguna melakukan transaksi lebih dari satu kali. Setelah satu transaksi selesai, pengguna diberikan pilihan untuk melanjutkan belanja atau mengakhiri program. Jika pengguna memilih melanjutkan, program akan kembali ke proses pemilihan barang. Perulangan ini membuat program menjadi lebih interaktif dan menyerupai sistem kasir yang digunakan pada toko sebenarnya. Secara keseluruhan, hasil implementasi program kasir sederhana ini menunjukkan bahwa konsep dasar pemrograman seperti variabel, `array`, percabangan, dan perulangan dapat diterapkan secara efektif untuk

membangun sebuah aplikasi sederhana yang memiliki fungsi nyata. Program ini tidak hanya berfungsi sebagai sarana pembelajaran, tetapi juga memberikan gambaran nyata mengenai penerapan pemrograman dalam sistem transaksi penjualan.

4. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, serta pengujian yang telah dilakukan, dapat disimpulkan bahwa program kasir sederhana yang dikembangkan menggunakan bahasa pemrograman C++ telah berhasil memenuhi tujuan awal pembuatan sistem. Program ini mampu menjalankan fungsi-fungsi dasar sistem kasir, mulai dari menampilkan daftar barang beserta harga, menerima *input* pilihan barang dan jumlah pembelian dari pengguna, hingga melakukan perhitungan total harga secara otomatis. Seluruh proses perhitungan dilakukan secara terstruktur sehingga dapat meminimalkan kesalahan yang biasanya terjadi pada perhitungan manual. Hasil pengujian menunjukkan bahwa program dapat menangani proses transaksi dengan baik, baik untuk satu kali pembelian maupun untuk beberapa kali pembelian dalam satu kali eksekusi program. Penerapan struktur perulangan memungkinkan pengguna untuk memilih apakah ingin melanjutkan proses belanja atau mengakhiri transaksi. Dengan adanya mekanisme tersebut, alur program menjadi lebih fleksibel dan menyerupai sistem kasir sederhana yang digunakan pada lingkungan toko secara nyata. Selain itu, seluruh subtotal dari setiap transaksi berhasil dijumlahkan dengan benar untuk menghasilkan total keseluruhan belanja yang harus dibayarkan oleh pengguna.

Dari sisi logika program, sistem ini telah menerapkan aturan pemberian diskon berdasarkan jenis barang dan jumlah pembelian secara tepat. Penggunaan struktur percabangan memungkinkan program untuk mengambil keputusan secara otomatis sesuai dengan kondisi yang telah ditentukan. Dengan demikian, proses perhitungan diskon dapat dilakukan secara konsisten dan transparan, sehingga pengguna dapat langsung mengetahui potongan harga yang diperoleh beserta subtotal akhir dari setiap transaksi. Dari sisi teknis pemrograman, program kasir sederhana ini telah menerapkan konsep dasar bahasa pemrograman C++ secara optimal. Konsep variabel digunakan untuk menyimpan data *input* dan hasil perhitungan, *array* dimanfaatkan untuk mengelola daftar barang dan harga secara efisien, struktur percabangan digunakan untuk menentukan alur keputusan, serta struktur perulangan diterapkan untuk mengatur proses transaksi berulang. Kombinasi dari konsep-konsep tersebut menghasilkan alur program yang sistematis, mudah dipahami, dan mudah dikembangkan lebih lanjut. Secara keseluruhan, program kasir sederhana ini mampu membantu proses transaksi penjualan menjadi lebih efektif dan efisien dibandingkan dengan metode perhitungan manual. Program ini tidak hanya berfungsi sebagai alat bantu transaksi, tetapi juga sebagai sarana pembelajaran yang baik bagi mahasiswa dalam memahami penerapan konsep dasar pemrograman C++ pada studi kasus nyata. Dengan adanya program ini, diharapkan mahasiswa dapat lebih memahami hubungan antara teori pemrograman dan implementasinya dalam bentuk aplikasi sederhana yang memiliki manfaat praktis.

REFERENCES

- [1] S. Alfarizi, Y. Sumaryana, and S. S. Sundari, "RANCANG BANGUN SISTEM INFORMASI JUAL BELI MOTOR MENGGUNAKAN ONE TIME PASSWORD (OTP) DAN MAILTRAP API," *J. Inform. Dan Tek. Elektro Terap.*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4137.
- [2] A. R. Pahlevi and M. Malabay, "Rancang Bangun Manajemen Sistem Informasi Jual Beli Produk Nugget (Studi Kasus: Toko Nugget Taman Narogong)," *IKRA-ITH Inform. J. Komput. Dan Inform.*, vol. 8, no. 2, pp. 22–33, July 2024, doi: 10.37817/ikraith-informatika.v8i2.2953.
- [3] K. Yasdomi, U. Utami, H. Maradona, D. Dona, and S. Rahayu, "Sistem Informasi Jual Beli Kelapa Sawit Berbasis Web pada Peron Reskianto," *INFORMATIKA*, vol. 12, no. 3, pp. 471–479, Dec. 2024, doi: 10.36987/informatika.v12i3.6117.
- [4] G. S. Putra, I. I. Maulana, A. D. Chayo, M. I. Haekal, and R. Syaharani, "Pengukuran Efektivitas Platform E-Learning dalam Pembelajaran Teknik Informatika di Era Digital," *J. MENTARI Manaj. Pendidik. Dan Teknol. Inf.*, vol. 3, no. 1, pp. 19–29, July 2024, doi: 10.33050/mentari.v3i1.559.
- [5] Arief Budiman, Y. D. Lestari, and M. Eka, "Penerapan Metode MAUT dalam Pemilihan Peminatan pada Program Studi Teknik Informatika," *J. UNITEK*, vol. 17, no. 2, pp. 169–180, Aug. 2024, doi: 10.52072/unitek.v17i2.921.
- [6] F. Domingo, J. Günther, J. S. Kim, and Z. S. Wang, "A C++ program for estimating detector sensitivities to long-lived particles: displaced decay counter," *Eur. Phys. J. C*, vol. 84, no. 6, p. 642, June 2024, doi: 10.1140/epjc/s10052-024-13009-9.
- [7] Y.-H. Chang, Y.-C. Yan, and Y.-T. Lu, "Effects of Combining Different Collaborative Learning Strategies with Problem-Based Learning in a Flipped Classroom on Program Language Learning," *Sustainability*, vol. 14, no. 9, p. 5282, Apr. 2022, doi: 10.3390/su14095282.

- [8] Y. Chen Kuang Piao, N. Ezzati-jivan, and M. R. Dagenais, "Distributed Architecture for an Integrated Development Environment, Large Trace Analysis, and Visualization," *Sensors*, vol. 21, no. 16, p. 5560, Aug. 2021, doi: 10.3390/s21165560.
- [9] Muh. Nur Rahmat, "Pemanfaatan Model Pembelajaran Project Based Learning (PjBL) untuk Mengasah Kreativitas dan Meningkatkan Hasil Belajar Siswa pada Mata Pelajaran Dasar Pemrograman," *J. Mediat.*, pp. 17–22, Sept. 2024, doi: 10.59562/mediatik.v7i3.4426.
- [10] H. D. Hermawan *et al.*, "Pelatihan Computational Thinking bagi Mahasiswa melalui CSUnplugged sebagai Penguatan Dasar-Dasar Pemrograman," *Lambung Inov. J. Pengabd. Kpd. Masy.*, vol. 9, no. 3, pp. 455–466, Sept. 2024, doi: 10.36312/linov.v9i3.2001.
- [11] F. Syahputra, E. Sabrina, T. Mediansyah, G. Armando, A. Sinaga, and Y. Perangin Angin, "Meta Analisis: Pengembangan Media Pembelajaran Dasar-Dasar Pemrograman Berbasis Desktop," *AR-RUMMAN J. Educ. Learn. Eval.*, vol. 1, no. 2, pp. 198–209, Nov. 2024, doi: 10.57235/arrumman.v1i2.3966.
- [12] E. N. Amini and B. Sujatmiko, "Rancang Bangun LMS Berbasis Web Mengimplementasikan Project Based Learning Untuk Meningkatkan Kompetensi Dasar-Dasar Pengembangan Perangkat Lunak dan Gim Pada Siswa Program Keahlian RPL (Studi Kasus Siswa Kelas X RPL di SMKN 10 Surabaya)," *IT-Edu J. Inf. Technol. Educ.*, vol. 9, no. 2, pp. 199–207, July 2024, doi: 10.26740/it-edu.v9i2.62495.
- [13] T. D. Andini, J. Arifin, . S., A. E. Irsyada, and R. D. Indahsari, "PELATIHAN PEMROGRAMAN BAHASA PYTHON PADA JURUSAN PERANGKAT LUNAK DAN GIM SMKN 12 MALANG," *J. Pengabd. Masy. - Teknol. Digit. Indones.*, vol. 2, no. 2, p. 42, Oct. 2023, doi: 10.26798/jpm.v2i2.880.
- [14] Z. Chen, Y. Zhu, and Z. Wang, "Design and Implementation of an Aspect-Oriented C Programming Language," *Proc. ACM Program. Lang.*, vol. 8, no. OOPSLA1, pp. 642–669, Apr. 2024, doi: 10.1145/3649834.
- [15] T. M. M. -, "C Programming: Structured Programming Language," *Int. J. Multidiscip. Res.*, vol. 6, no. 6, p. 30657, Nov. 2024, doi: 10.36948/ijfmr.2024.v06i06.30657.
- [16] A. Alagarsamy, "C Programming Language Better Understanding," *Int. J. Multidiscip. Res.*, vol. 7, no. 3, p. 49293, June 2025, doi: 10.36948/ijfmr.2025.v07i03.49293.
- [17] B. A. Saputra, S. C. Candra, F. B. Wijaya, K. M. Suryaningrum, and H. A. Saputri, "Comparative Analysis of Binary and Interpolation Search Algorithms on Integer Data Using C Programming Language," *2023 Int. Conf. Inf. Manag. Technol. ICIMTech*, pp. 340–345, Aug. 2023, doi: 10.1109/ICIMTech59029.2023.10277955.
- [18] I. I. Petrila, "@C -- augmented version of C programming language," 2022, doi: 10.48550/ARXIV.2212.11245.
- [19] A. I. Legalov, Y. G. Bugayenko, N. K. Chuykin, M. V. Shipitsin, Y. I. Riabtsev, and A. N. Kamenskiy, "Transformation of C Programming Language Memory Model into Object-Oriented Representation of EO Language," *Autom. Control Comput. Sci.*, vol. 57, no. 7, pp. 803–816, Dec. 2023, doi: 10.3103/S0146411623070088.
- [20] S. Peta, "C Programming Language–Still Ruling the World," *Glob. J. Comput. Sci. Technol.*, pp. 1–5, July 2022, doi: 10.34257/GJCSTHVOL22IS1PG1.
- [21] S. B. Uzayr, "Mastering C++ Programming Language: A Beginner's Guide," Boca Raton: CRC Press, Feb. 2022. doi: 10.1201/9781003214762.