

LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN LANJUT

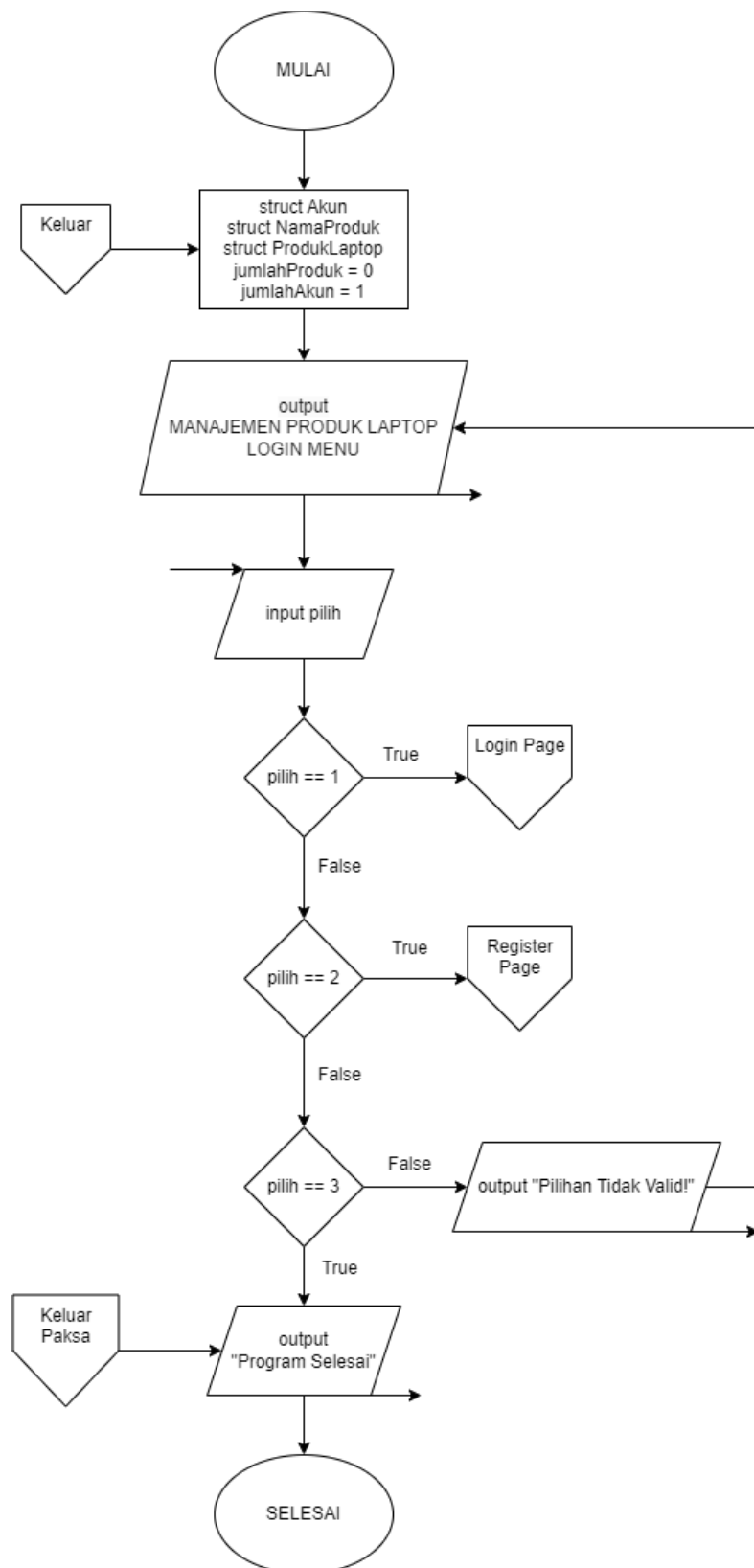


Disusun oleh:
Zeydan Fazle Mawla (2409106010)
Kelas (A1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

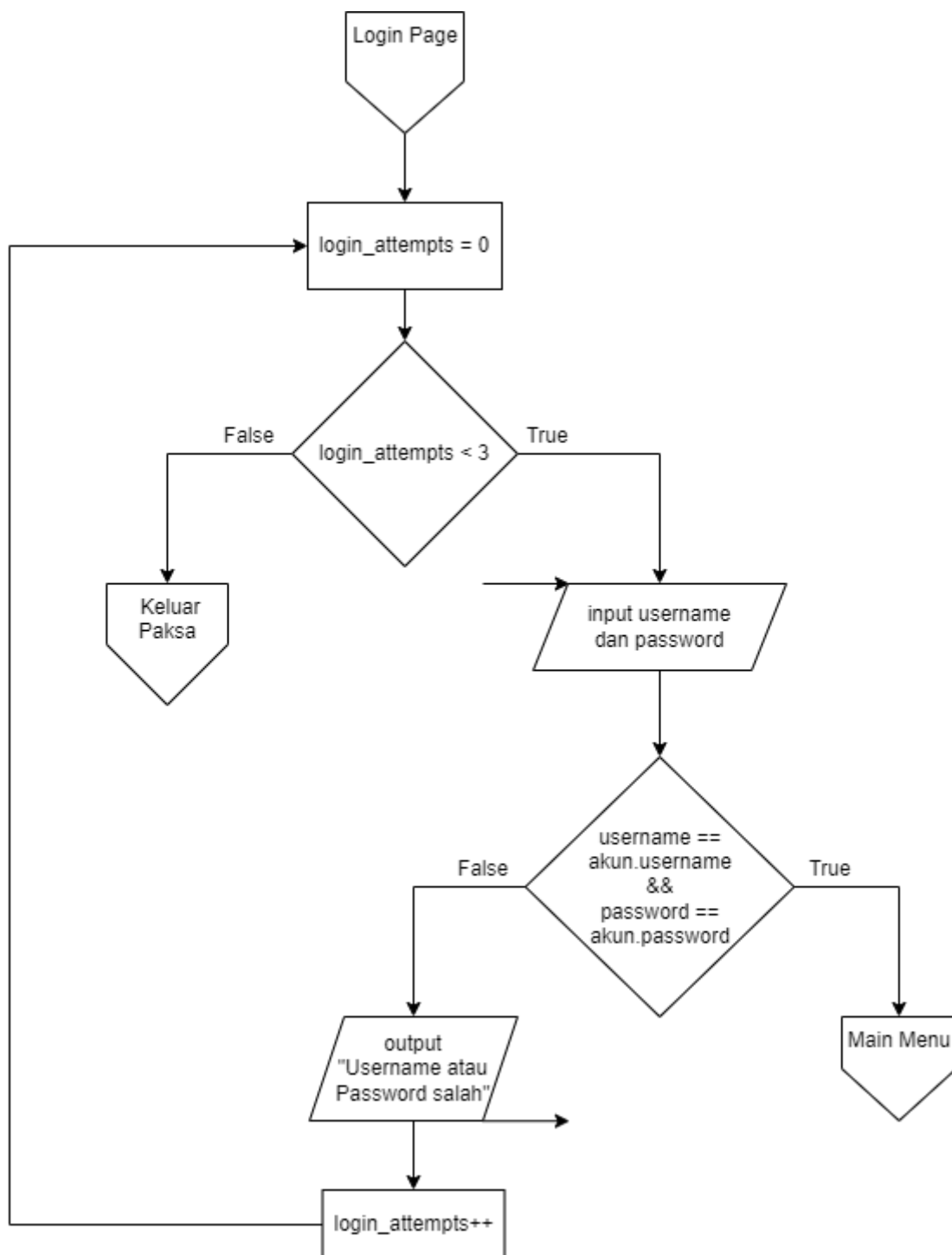
1. Flowchart

1.1 Menu Login



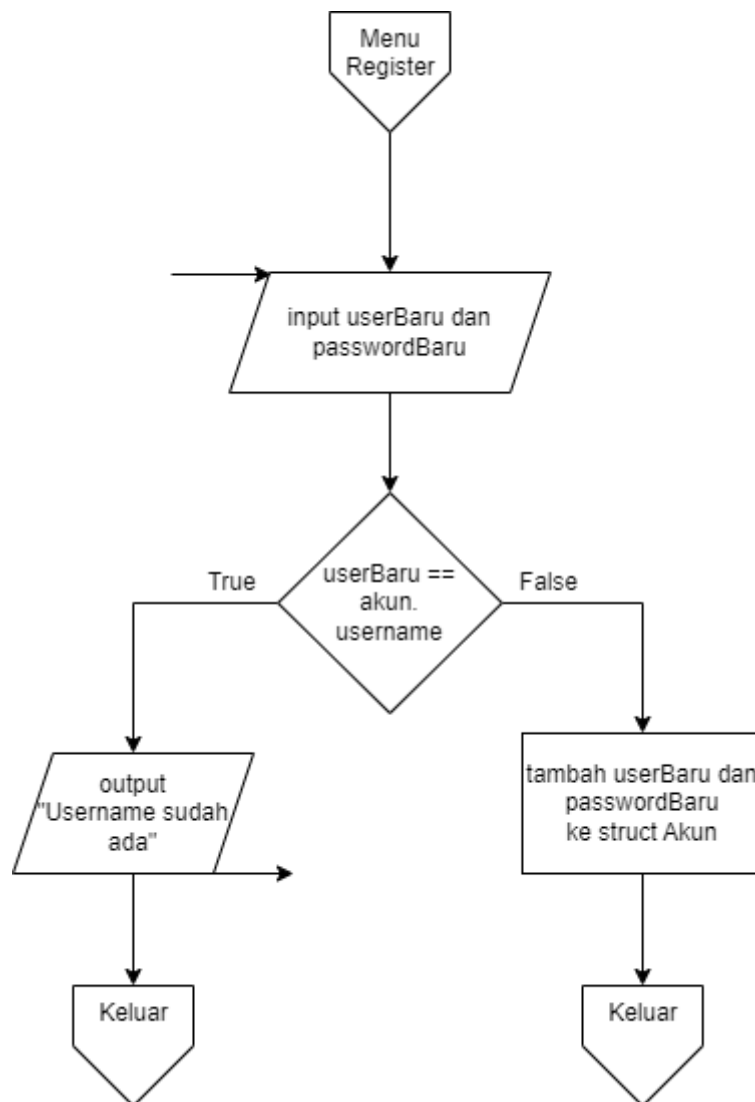
Gambar 1.1 Menu Login

1.2 Menu Validasi Login



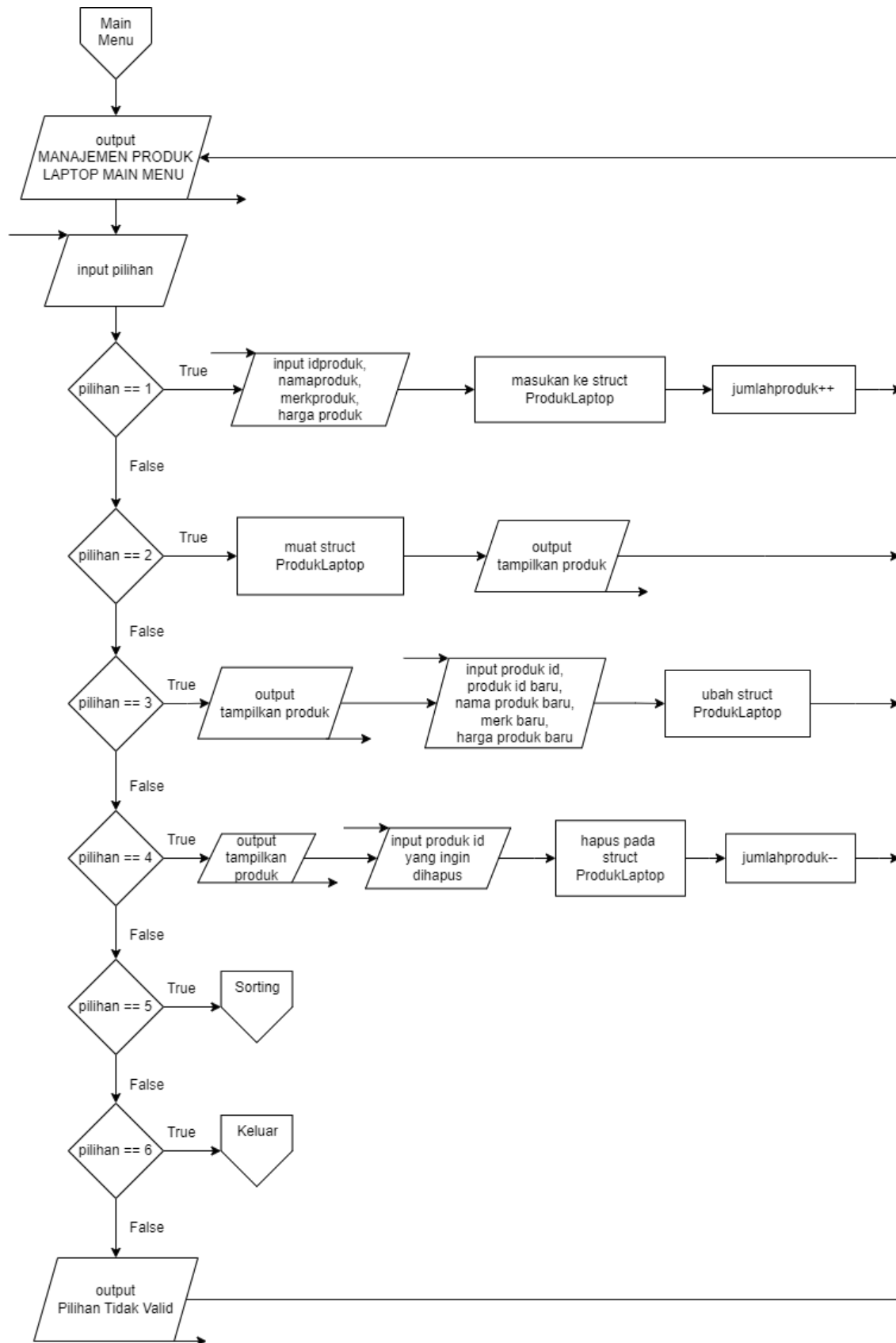
Gambar 1.2 Menu Validasi Login

1.3 Menu Register



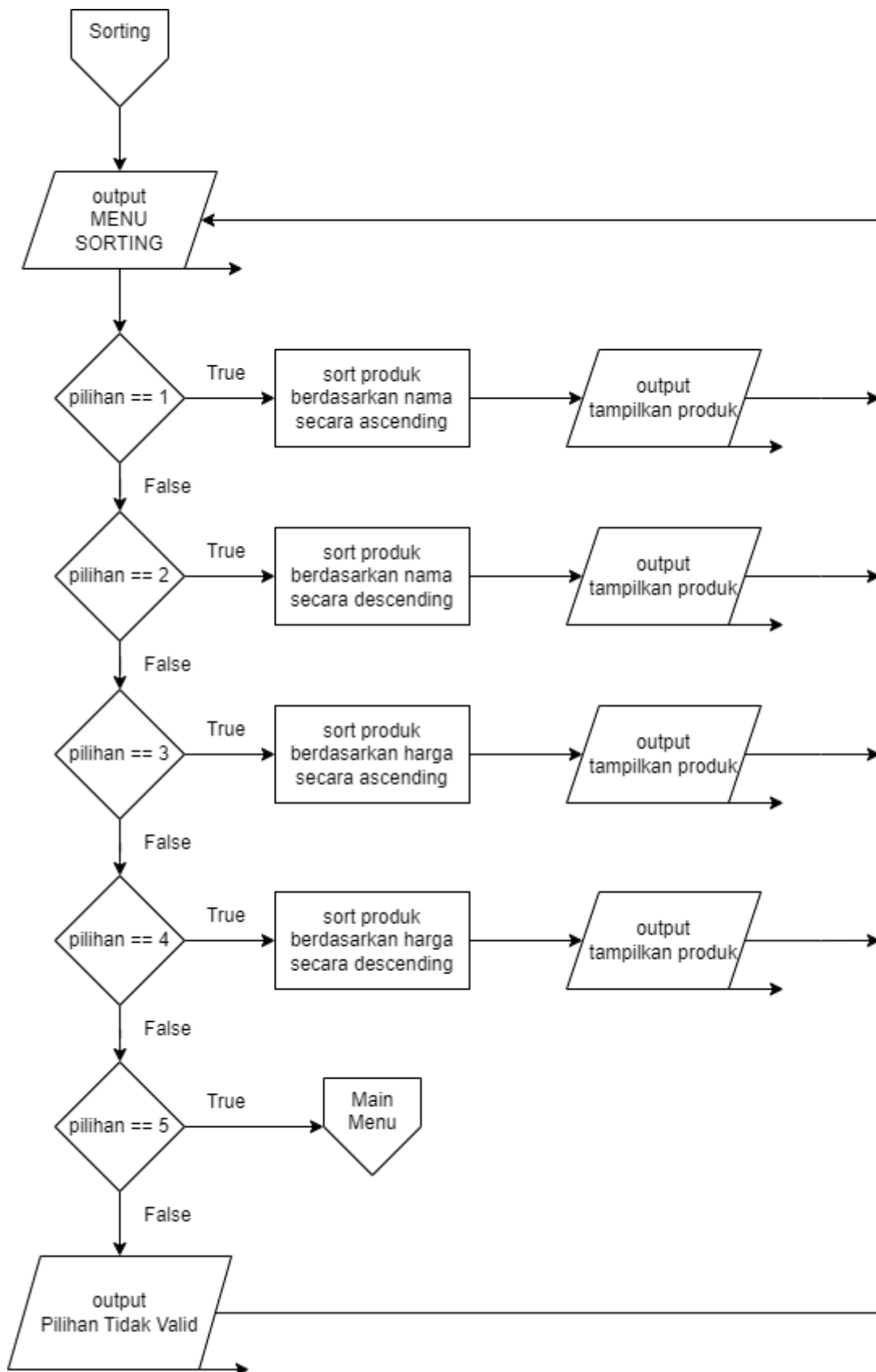
Gambar 1.3 Menu Register

1.4 Menu Utama



Gambar 1.4 Menu Utama

1.5 Menu Sorting



Gambar 1.5 Menu Sorting

2. Analisis Program

2.1 Tujuan Program

Program dibuat untuk manajemen produk laptop yang dapat menambah laptop, melihat daftar laptop, mengubah laptop, dan menghapus laptop setelah pengguna berhasil register dan login ke menu utama. Data produk yang ditambahkan disimpan kedalam masing-masing struct yang memuat, id, nama, merek dan harga. Beberapa fungsi menggunakan menerapkan rekursif, overloading, dan pointer.

2.2 Fungsi Program

1. Manajemen Akun
 - a. Pengguna dapat login menggunakan akun yang telah terdaftar
 - b. Pengguna dapat mendaftar akun baru
2. Manajemen Produk Laptop
 - a. Menambahkan laptop baru dengan ID, merek, nama, dan harga
 - b. Melihat daftar produk dalam bentuk tabel
 - c. Mengubah informasi produk berdasarkan ID yang dipilih
 - d. Menghapus produk dari daftar
 - e. Mengurutkan data secara ascending dan descending berdasarkan nama dan harga
3. Validasi
 - a. Pembatasan percobaan login maksimal 3 kali
 - b. Validasi ID produk agar tidak duplikat
 - c. Validasi username agar tidak duplikat

3. Source Code

A. Fitur Menu Login

Fitur ini digunakan untuk menampilkan menu login dan register

Source Code:

```
void menuLogin () {  
    cout << "\n=====\\n";  
    cout << "|   MANAJEMEN PRODUK LAPTOP   |\\n";  
    cout << "=====\\n";  
    cout << "| 1. Login                     |\\n";  
    cout << "| 2. Register                  |\\n";  
    cout << "| 3. Keluar                    |\\n";  
    cout << "=====\\n";  
    cout << "Masukkan Pilihan: ";  
}
```

B. Fitur Validasi Login

Fitur ini digunakan untuk input username dan password dan memvalidasinya agar bisa masuk ke program utama

Source Code:

```
void login() {  
    clear();  
    string username, password;  
    int login_attempts = 0;  
    bool login_success = false;  
  
    while (login_attempts < 3) {  
        cout << "\\n=====\\n";  
        cout << "\\n|                               LOGIN                               |\\n";  
        cout << "\\n=====\\n";  
        cout << "\\nMasukkan Username: ";  
        cin >> username;  
        cout << "Masukkan Password: ";  
        cin >> password;  
  
        bool found = false;  
        for (int i = 0; i < jumlahAkun; i++) {  
            if (daftarAkun[i].username == username && daftarAkun[i].password ==  
password) {  
                found = true;  
                break;  
            }  
        }  
    }  
}
```



```

    }
}

if (found) {
    login_success = true;
    cout << "Login Berhasil.\n";
    break;
} else {
    cout << "Username atau Password salah! Percobaan ke-" <<
login_attempts + 1 << " dari 3\n";
    login_attempts++;
}
}

if (!login_success) {
    cout << "Kamu gagal login 3 kali. Program berhenti." << endl;
    loginFailed = true;
    return;
}

string pilihan;
while (login_success) {
    menuUtama();
    cin >> pilihan;
    cin.ignore();
    clear();

    if (pilihan == "1") {
        tambahLaptop();
    } else if (pilihan == "2") {
        int index = 0;
        lihatLaptop(index);
    } else if (pilihan == "3") {
        ubahLaptop();
    } else if (pilihan == "4") {
        hapusLaptop();
    } else if (pilihan == "5") {
        sortingLaptop();
    } else if (pilihan == "6") {
        login_success = false;
    } else {
        cout << "Pilihan tidak valid!\n";
    }
}
}
}

```

C. Fitur Menu Register

Fitur ini digunakan untuk membuat akun baru

Source Code:

```
void registerAkun() {
    clear();
    string userBaru;
    if (jumlahAkun >= MAX_AKUN) {
        cout << "Jumlah akun maksimal telah tercapai!" << endl;
    } else {
        cout << "\n=====";
        cout << "\n|          REGISTER          |";
        cout << "\n=====";
        cout << "\nMasukkan Username: ";
        cin >> userBaru;
        bool userExists = false;
        for (int i = 0; i < jumlahAkun; i++) {
            if (daftarAkun[i].username == userBaru) {
                cout << "Username sudah ada! Gunakan user lain." << endl;
                userExists = true;
                break;
            }
        }
        if (!userExists) {
            daftarAkun[jumlahAkun].username = userBaru;
            cout << "Masukkan Password: ";
            cin >> daftarAkun[jumlahAkun].password;
            jumlahAkun++;
            cout << "Akun berhasil didaftarkan!" << endl;
        }
    }
}
```

D. Fitur Menu Utama

Fitur ini digunakan untuk menampilkan menu utama

Source Code:

```
void menuUtama() {
    cout << "\n=====\\n";
    cout << "|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |\\n";
    cout << "=====\\n";
    cout << "|  1. Tambah Laptop                    |\\n";
    cout << "|  2. Lihat Laptop                    |\\n";
    cout << "|  3. Ubah Laptop                     |\\n";
    cout << "|  4. Hapus Laptop                    |\\n";
    cout << "|  5. Logout                          |\\n";
}
```

```

    cout << "=====\n";
    cout << "Masukkan Pilihan: ";
}

```

E. Fitur Tambah Laptop

Fitur ini digunakan untuk menambahkan produk laptop dan memastikan tidak ada id yang sama

Source Code:

```

bool idExists(int id) {
    for (int i = 0; i < jumlahProduk; i++) {
        if (daftarProduk[i].id == id) {
            return true;
        }
    }
    return false;
}

void tambahLaptop() {
    if (jumlahProduk >= MAX_PRODUK) {
        cout << "Data penuh! Tidak bisa menambahkan produk lagi." << endl;
        return;
    }

    int id, harga;
    string merek, nama;

    cout << "Masukkan ID Produk: ";
    cin >> id;
    if (idExists(id)) {
        cout << "ID sudah digunakan! Gunakan ID lain.\n";
        return;
    }

    cout << "Masukkan Harga Produk: ";
    cin >> harga;
    cin.ignore();

    cout << "Masukkan Merek Produk: ";
    getline(cin, merek);
    cout << "Masukkan Nama Produk: ";
    getline(cin, nama);

    if (tambahLaptop(&id, &harga) && tambahLaptop(&merek, &nama)) {
        cout << "Produk berhasil ditambahkan!\n";
    } else {
        cout << "Gagal menambahkan produk.\n";
    }
}

```

```

    }
}

bool tambahLaptop(int* id, int* harga) {
    if (jumlahProduk >= MAX_PRODUK) return false;
    daftarProduk[jumlahProduk].id = *id;
    daftarProduk[jumlahProduk].harga = *harga;
    return true;
}

bool tambahLaptop(string* merek, string* nama) {
    if (jumlahProduk >= MAX_PRODUK) return false;
    daftarProduk[jumlahProduk].namaProduk.merek = *merek;
    daftarProduk[jumlahProduk].namaProduk.nama = *nama;
    jumlahProduk++;
    return true;
}

```

F. Fitur Lihat Laptop

Fitur ini digunakan untuk menampilkan produk laptop dalam bentuk tabel

Source Code:

```

bool lihatLaptop(int& index) {
    if (jumlahProduk == 0) {
        cout << "Tidak ada produk.\n";
        return false;
    }

    if (index == 0) {
        cout << "\nDaftar Produk:\n";
        cout << left << setw(10) << "ID"
            << setw(20) << "Merek"
            << setw(25) << "Nama"
            << setw(15) << "Harga" << endl;
        cout <<
            "-----\n";
    }

    if (index >= jumlahProduk) {
        return true;
    }

    cout << left << setw(10) << daftarProduk[index].id
        << setw(20) << daftarProduk[index].namaProduk.merek
        << setw(25) << daftarProduk[index].namaProduk.nama
        << "Rp " << daftarProduk[index].harga << endl;
}

```

```

        index++;
        return lihatLaptop(index);
    }

```

G. Fitur Ubah Laptop

Fitur ini digunakan untuk mengubah produk laptop dan memastikan tidak ada id yang sama

Source Code:

```

void ubahLaptop() {
    int index = 0;
    if (jumlahProduk == 0) {
        cout << "Tidak ada produk yang dapat diubah!\n";
    } else {
        lihatLaptop(index);
        int id, index = -1;
        cout << "\nMasukkan ID produk yang ingin diubah: ";
        cin >> id;
        cin.ignore();

        for (int i = 0; i < jumlahProduk; i++) {
            if (daftarProduk[i].id == id) {
                index = i;
                break;
            }
        }
        if (index != -1) {
            int idBaru;
            cout << "Masukkan ID baru: ";
            cin >> idBaru;

            bool idSudahAda = false;
            for (int i = 0; i < jumlahProduk; i++) {
                if (daftarProduk[i].id == idBaru && i != index) {
                    cout << "ID baru sudah digunakan oleh produk lain!\n";
                    idSudahAda = true;
                    break;
                }
            }

            if (!idSudahAda) {
                daftarProduk[index].id = idBaru;
                cin.ignore();
                cout << "Masukkan Merek baru: ";
                getline(cin, daftarProduk[index].namaProduk.merek);
                cout << "Masukkan Nama baru: ";
                getline(cin, daftarProduk[index].namaProduk.nama);
                cout << "Masukkan Harga baru: ";
            }
        }
    }
}

```

```

        cin >> daftarProduk[index].harga;
        cin.ignore();
        cout << "Produk berhasil diperbarui!\n";
    }
} else {
    cout << "Produk tidak ditemukan!\n";
}
}
}
}

```

H. Fitur Hapus Laptop

Fitur ini digunakan untuk menghapus produk laptop yang telah ditambahkan

Source Code:

```

void hapusLaptop() {
    int index = 0;
    if (jumlahProduk == 0) {
        cout << "Tidak ada produk yang dapat dihapus!\n";
    } else {
        lihatLaptop(index);
        int id, index = -1;
        cout << "\nMasukkan ID produk yang ingin dihapus: ";
        cin >> id;
        cin.ignore();

        for (int i = 0; i < jumlahProduk; i++) {
            if (daftarProduk[i].id == id) {
                index = i;
                break;
            }
        }

        if (index != -1) {
            for (int i = index; i < jumlahProduk - 1; i++) {
                daftarProduk[i] = daftarProduk[i + 1];
            }

            jumlahProduk--;
            cout << "Produk berhasil dihapus!\n";
        } else {
            cout << "Produk tidak ditemukan!\n";
        }
    }
}
}

```

I. Fitur Bersihkan Terminal

Fitur ini digunakan untuk membersihkan terminal

Source Code:

```
void clear() {
    system("cls");
}
```

J. Fitur Menu Sorting

Fitur ini digunakan untuk menampilkan menu sorting

Source Code:

```
void sortingLaptop() {
    while (true) {
        if (jumlahProduk == 0) {
            cout << "Tidak ada produk untuk di-sort.\n";
            return;
        }

        cout << "\n===== \n";
        cout << " |          MENU SORTING          | \n";
        cout << "===== \n";
        cout << " | 1. Berdasarkan Nama Ascending      | \n";
        cout << " | 2. Berdasarkan Nama Descending     | \n";
        cout << " | 3. Berdasarkan Harga Ascending     | \n";
        cout << " | 4. Berdasarkan Harga Descending    | \n";
        cout << " | 5. Kembali                        | \n";
        cout << "===== \n";
        cout << "Masukkan Pilihan: ";

        int pilihan;
        cin >> pilihan;
        cin.ignore();

        switch (pilihan) {
            case 1:
                sortLaptop(daftarProduk, 0, jumlahProduk - 1, true);
                cout << "Produk diurutkan berdasarkan nama (ascending).\n";
                break;
            case 2:
                sortLaptop(daftarProduk, 0, jumlahProduk - 1, false);
                cout << "Produk diurutkan berdasarkan nama (descending).\n";
                break;
            case 3:
                sortHarga(daftarProduk, 0, jumlahProduk - 1, true);
                cout << "Produk diurutkan berdasarkan harga (ascending).\n";
```

```

        break;
    case 4:
        sortHarga(daftarProduk, 0, jumlahProduk - 1, false);
        cout << "Produk diurutkan berdasarkan harga (descending).\n";
        break;
    case 5:
        clear();
        return;
    default:
        cout << "Pilihan tidak valid!\n";
        continue;
}

int index = 0;
lihatLaptop(index);
}
}

```

K. Fitur Sorting

Fitur ini digunakan untuk mengurutkan data berdasarkan nama dan harga

Source Code:

```

void swapProduk(ProdukLaptop &a, ProdukLaptop &b) {
    ProdukLaptop temp = a;
    a = b;
    b = temp;
}

void sortHarga(ProdukLaptop arr[], int low, int high, bool ascending) {
    if (low < high) {
        int pivot = low;
        int i = low;
        int j = high;
        while (i <= j) {
            if (ascending) {
                while (arr[i].harga <= arr[pivot].harga && i <= high) i++;
                while (arr[j].harga > arr[pivot].harga && j >= low) j--;
            } else {
                while (arr[i].harga >= arr[pivot].harga && i <= high) i++;
                while (arr[j].harga < arr[pivot].harga && j >= low) j--;
            }
            if (i < j) swapProduk(arr[i], arr[j]);
        }
        swapProduk(arr[j], arr[pivot]);
        sortHarga(arr, low, j - 1, ascending);
        sortHarga(arr, j + 1, high, ascending);
    }
}

```



```

}

void sortLaptop(ProdukLaptop arr[], int low, int high, bool ascending) {
    if (low < high) {
        int pivot = low;
        int i = low;
        int j = high;
        while (i <= j) {
            if (ascending) {
                while (arr[i].namaProduk.nama <= arr[pivot].namaProduk.nama && i
<= high) i++;
                while (arr[j].namaProduk.nama > arr[pivot].namaProduk.nama && j
>= low) j--;
            } else {
                while (arr[i].namaProduk.nama >= arr[pivot].namaProduk.nama && i
<= high) i++;
                while (arr[j].namaProduk.nama < arr[pivot].namaProduk.nama && j
>= low) j--;
            }
            if (i < j) swapProduk(arr[i], arr[j]);
        }
        swapProduk(arr[j], arr[pivot]);
        sortLaptop(arr, low, j - 1, ascending);
        sortLaptop(arr, j + 1, high, ascending);
    }
}

```

4. Uji Coba dan Hasil Output

```
=====
|   MANAJEMEN PRODUK LAPTOP   |
=====
| 1. Login                    |
| 2. Register                 |
| 3. Keluar                   |
=====
Masukkan Pilihan: █
```

Gambar 4.1 Menu Awal

```
=====
|           LOGIN           |
=====
Masukkan Username: zeydanfazlemawla
Masukkan Password: 2409106010
Login Berhasil.

=====
| MANAJEMEN PRODUK LAPTOP MAIN MENU |
=====
| 1. Tambah Laptop           |
| 2. Lihat Laptop            |
| 3. Ubah Laptop             |
| 4. Hapus Laptop            |
| 5. Logout                  |
=====
Masukkan Pilihan: █
```

Gambar 4.2 Login Berhasil dan Masuk Menu Utama

```
=====
|             LOGIN             |
=====
Masukkan Username: wahyu
Masukkan Password: 123
Username atau Password salah! Percobaan ke-1 dari 3

=====
|             LOGIN             |
=====
Masukkan Username: casto
Masukkan Password: rateoff
Username atau Password salah! Percobaan ke-2 dari 3

=====
|             LOGIN             |
=====
Masukkan Username: tolak
Masukkan Password: globalpassive
Username atau Password salah! Percobaan ke-3 dari 3
Kamu gagal login 3 kali. Program berhenti.
PS D:\praktikum-apl\post-test\post-test-4>
```

Gambar 4.3 Program Terhenti Karena Login Gagal

```

=====
|          REGISTER          |
=====
Masukkan Username: alucard
Masukkan Password: miya
Akun berhasil didaftarkan!

=====
|  MANAJEMEN PRODUK LAPTOP  |
=====
| 1. Login                  |
| 2. Register               |
| 3. Keluar                 |
=====
Masukkan Pilihan: █

```

Gambar 4.4 Register

```

=====
|          REGISTER          |
=====
Masukkan Username: alucard
Username sudah ada! Gunakan user lain.

=====
|  MANAJEMEN PRODUK LAPTOP  |
=====
| 1. Login                  |
| 2. Register               |
| 3. Keluar                 |
=====
Masukkan Pilihan: █

```

Gambar 4.5 Register dengan Username yang Sama

```
=====
|                               LOGIN                               |
=====
Masukkan Username: alucard
Masukkan Password: miya
Login Berhasil.

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
| 1. Tambah Laptop |
| 2. Lihat Laptop  |
| 3. Ubah Laptop   |
| 4. Hapus Laptop  |
| 5. Logout        |
=====
Masukkan Pilihan: █
```

Gambar 4.6 Login dengan Akun Baru

```
Masukkan ID Produk: 1010
Masukkan Harga Produk: 14000000
Masukkan Merek Produk: ASUS
Masukkan Nama Produk: VIVOB00K
Produk berhasil ditambahkan!

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
| 1. Tambah Laptop |
| 2. Lihat Laptop  |
| 3. Ubah Laptop   |
| 4. Hapus Laptop  |
| 5. Logout        |
=====
Masukkan Pilihan: █
```

Gambar 4.7 Menu Tambah Laptop

```
Daftar Produk:
ID          Merek          Nama          Harga
-----
1010        ASUS            VIVOB00K      Rp 140000000

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: █
```

Gambar 4.8 Menu Lihat Laptop

```
Daftar Produk:
ID          Merek          Nama          Harga
-----
1010        ASUS            VIVOB00K      Rp 140000000

Masukkan ID produk yang ingin diubah: 1010
Masukkan ID baru: 666
Masukkan Merek baru: LENOVO
Masukkan Nama baru: YOGA
Masukkan Harga baru: 250000000
Produk berhasil diperbarui!

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: █
```

Gambar 4.9 Menu Ubah Laptop

```

Daftar Produk:
ID          Merek          Nama          Harga
-----
666         LENOVO          YOGA          Rp 25000000

Masukkan ID produk yang ingin dihapus: 666
Produk berhasil dihapus!

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: █

```

Gambar 4.10 Menu Hapus Laptop

```

Masukkan ID Produk: 11
ID sudah digunakan! Gunakan ID lain.

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: █

```

Gambar 4.11 Tambah Laptop dengan ID yang Sama

```

Tidak ada produk.

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: █

```

Gambar 4.12 Produk Kosong Ketika Melihat Laptop

```

Daftar Produk:
ID      Merek      Nama      Harga
-----
123     AXIO      PONGO     Rp 12000000

Masukkan ID produk yang ingin diubah: 666
Produk tidak ditemukan!

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
|  1. Tambah Laptop                    |
|  2. Lihat Laptop                     |
|  3. Ubah Laptop                      |
|  4. Hapus Laptop                     |
|  5. Logout                           |
=====
Masukkan Pilihan: █

```

Gambar 4.13 Menginput ID yang tidak ada dalam Menu Ubah


```

Daftar Produk:
ID          Merek          Nama          Harga
-----
123         AXIO           PONGO        Rp 12000000

Masukkan ID produk yang ingin dihapus: 000
Produk tidak ditemukan!

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
| 1. Tambah Laptop                    |
| 2. Lihat Laptop                     |
| 3. Ubah Laptop                      |
| 4. Hapus Laptop                     |
| 5. Logout                          |
=====
Masukkan Pilihan: █

```

Gambar 4.14 Menginput ID yang Tidak Ada dalam Menu Hapus

```

Daftar Produk:
ID          Merek          Nama          Harga
-----
123         AXIO           PONGO        Rp 12000000
999         ASUS            ROG          Rp 25000000

Masukkan ID produk yang ingin diubah: 123
Masukkan ID baru: 999
ID baru sudah digunakan oleh produk lain!

=====
|  MANAJEMEN PRODUK LAPTOP MAIN MENU  |
=====
| 1. Tambah Laptop                    |
| 2. Lihat Laptop                     |
| 3. Ubah Laptop                      |
| 4. Hapus Laptop                     |
| 5. Logout                          |
=====
Masukkan Pilihan: █

```

Gambar 4.15 Menginput ID Baru yang Sudah Ada dalam Menu Ubah

```
=====
|                               MENU SORTING                               |
=====
| 1. Berdasarkan Nama Ascending |
| 2. Berdasarkan Nama Descending |
| 3. Berdasarkan Harga Ascending |
| 4. Berdasarkan Harga Descending |
| 5. Kembali                      |
=====
Masukkan Pilihan: █
```

Gambar 4.16 Menu Sorting

```
Produk diurutkan berdasarkan nama (ascending).

Daftar Produk:
ID      Merek      Nama      Harga
-----
109     Samsung     Galaxy Book     Rp 9000000
110     LG             Gram            Rp 11000000
106     Apple         MacBook Air     Rp 14000000
103     HP             Pavilion        Rp 8500000
107     MSI            Prestige        Rp 13000000
108     Toshiba       Satellite       Rp 5000000
105     Acer           Swift           Rp 6000000
102     Lenovo        ThinkPad        Rp 12000000
101     Asus           Vivobook        Rp 7500000
104     Dell           XPS             Rp 15000000

=====
|                               MENU SORTING                               |
=====
| 1. Berdasarkan Nama Ascending |
| 2. Berdasarkan Nama Descending |
| 3. Berdasarkan Harga Ascending |
| 4. Berdasarkan Harga Descending |
| 5. Kembali                      |
=====
Masukkan Pilihan: █
```

Gambar 4.17 Sort Nama Ascending

```
Produk diurutkan berdasarkan nama (descending).

Daftar Produk:
ID      Merek      Nama      Harga
-----
104     Dell      XPS      Rp 15000000
101     Asus      Vivobook  Rp 7500000
102     Lenovo    ThinkPad  Rp 12000000
105     Acer      Swift     Rp 6000000
108     Toshiba   Satellite Rp 5000000
107     MSI       Prestige  Rp 13000000
103     HP        Pavilion  Rp 8500000
106     Apple     MacBook Air Rp 14000000
110     LG        Gram      Rp 11000000
109     Samsung   Galaxy Book Rp 9000000

=====
|          MENU SORTING          |
=====
| 1. Berdasarkan Nama Ascending |
| 2. Berdasarkan Nama Descending |
| 3. Berdasarkan Harga Ascending |
| 4. Berdasarkan Harga Descending |
| 5. Kembali                     |
=====
Masukkan Pilihan: █
```

Gambar 4.18 Sort Nama Descending

```
Produk diurutkan berdasarkan harga (ascending).

Daftar Produk:
ID      Merek      Nama      Harga
-----
108     Toshiba   Satellite Rp 5000000
105     Acer      Swift     Rp 6000000
101     Asus      Vivobook  Rp 7500000
103     HP        Pavilion  Rp 8500000
109     Samsung   Galaxy Book Rp 9000000
110     LG        Gram      Rp 11000000
102     Lenovo    ThinkPad  Rp 12000000
107     MSI       Prestige  Rp 13000000
106     Apple     MacBook Air Rp 14000000
104     Dell      XPS      Rp 15000000

=====
|          MENU SORTING          |
=====
| 1. Berdasarkan Nama Ascending |
| 2. Berdasarkan Nama Descending |
| 3. Berdasarkan Harga Ascending |
| 4. Berdasarkan Harga Descending |
| 5. Kembali                     |
=====
Masukkan Pilihan: █
```

Gambar 4.19 Sort Harga Ascending

Produk diurutkan berdasarkan harga (descending).

Daftar Produk:

ID	Merek	Nama	Harga
104	Dell	XPS	Rp 15000000
106	Apple	MacBook Air	Rp 14000000
107	MSI	Prestige	Rp 13000000
102	Lenovo	ThinkPad	Rp 12000000
110	LG	Gram	Rp 11000000
109	Samsung	Galaxy Book	Rp 9000000
103	HP	Pavilion	Rp 8500000
101	Asus	Vivobook	Rp 7500000
105	Acer	Swift	Rp 6000000
108	Toshiba	Satellite	Rp 5000000

```
=====
|          MENU SORTING          |
=====
| 1. Berdasarkan Nama Ascending  |
| 2. Berdasarkan Nama Descending |
| 3. Berdasarkan Harga Ascending |
| 4. Berdasarkan Harga Descending|
| 5. Kembali                     |
=====
Masukkan Pilihan: 
```

Gambar 4.20 Sort Harga Descending

```
=====
|  MANAJEMEN PRODUK LAPTOP  |
=====
| 1. Login                  |
| 2. Register               |
| 3. Keluar                 |
=====
Masukkan Pilihan: 3
Program selesai.
PS D:\praktikum-apl\post-test\post-test-4> 
```

Gambar 4.21 Keluar Program

5. Langkah-Langkah Git pada VSCode

5.1 Git Init

```
● PS D:\praktikum-apl> git init
Reinitialized existing Git repository in D:/praktikum-apl/.git/
```

Gambar 5.1 Git Init

Command ini berfungsi untuk menginisiasi repositori yang ada pada file lokal dan berlokasi di folder .git

5.2 Git Add

```
● PS D:\praktikum-apl> git add .
```

Gambar 5.2 Git Add

Command ini berfungsi untuk menambah file yang ingin dicommit

5.3 Git Commit

```
● PS D:\praktikum-apl\post-test\post-test-6> git commit -m "add sorting"
[main 529caee] add sorting
3 files changed, 468 insertions(+)
○ create mode 100644 post-test/post-test-6/2409106010-ZeydanFazleMawla-PT-6.cpp
○ create mode 100644 post-test/post-test-6/2409106010-ZeydanFazleMawla-PT-6.exe
○ create mode 100644 post-test/post-test-6/2409106010-ZeydanFazleMawla-PT-6.pdf
```

Gambar 5.3 Git Commit

Command ini berfungsi untuk menyimpan perubahan ke repositori lokal git dengan pesan

5.4 Git Remote

```
● PS D:\praktikum-apl> git remote add origin https://github.com/wahyugod/praktikum-apl.git
```

Gambar 5.4 Git Remote

Command ini berfungsi untuk menghubungkan repository yang ada di lokal komputer dengan repository cloud pada github pada pertama kali

5.5 Git Push

```
● PS D:\praktikum-apl\post-test\post-test-6> git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1.04 MiB | 842.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/wahyugod/praktikum-apl.git
   0d5044e..529caee  main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.5 Git Push

Command ini berfungsi untuk mengupload semua hal yang ada di repository lokal ke github