

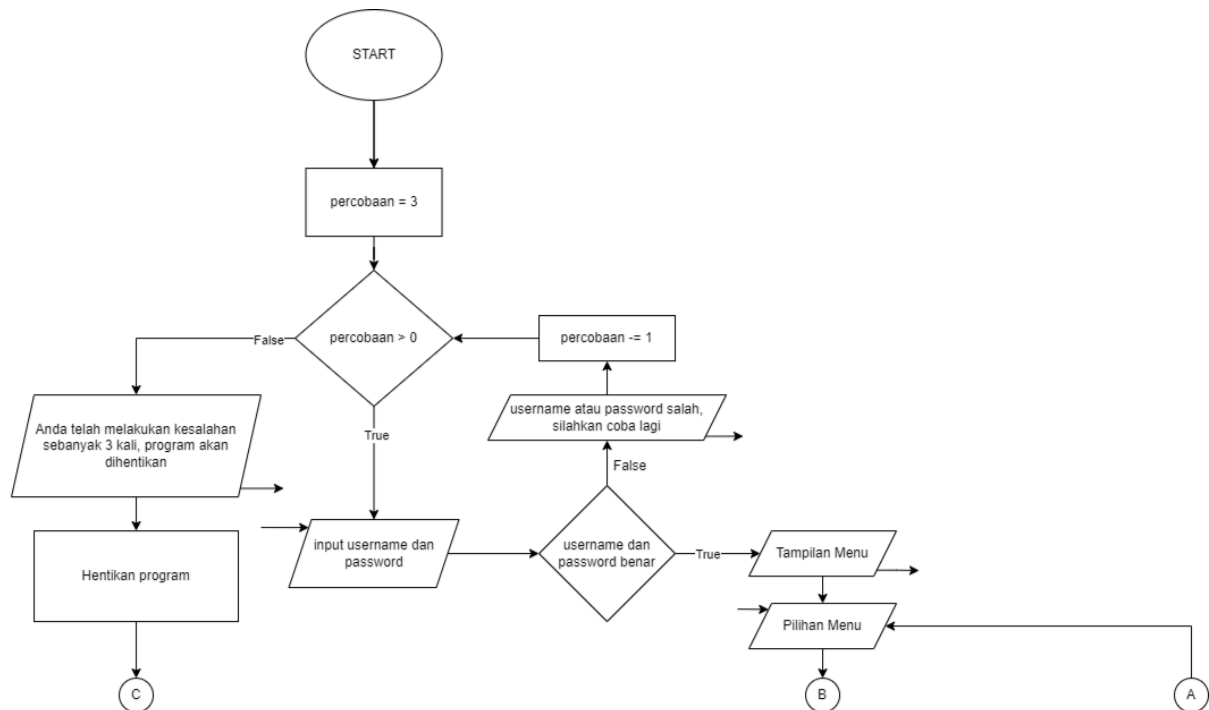
**LAPORAN PRAKTIKUM**  
**POSTTEST 4**  
**ALGORITMA PEMROGRAMAN LANJUT**



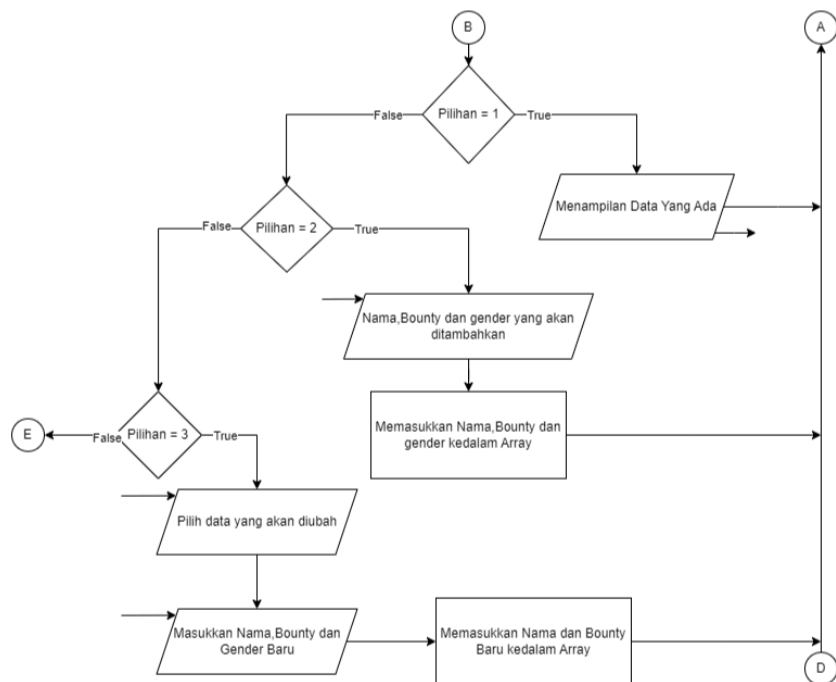
**Disusun oleh:**  
**Fariz Muwaffaq (2409106032)**  
**Kelas (A2 '24)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

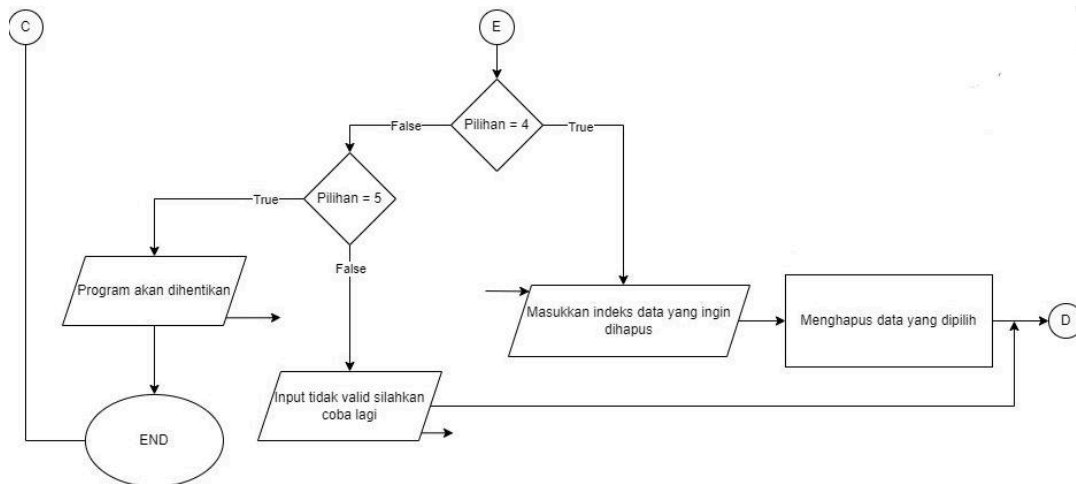
## 1. Flowchart



Gambar 1.1 Flowchart



Gambar 1.2 Flowchart



Gambar 1.3 Flowchart

## 2. Analisis Program

Program ini digunakan untuk membuat list karakter One Piece dan bounty-nya. Sebelum masuk ke dalam menu, pengguna diminta untuk login terlebih dahulu dan ketika pengguna salah memasukkan username atau password maka program akan berhenti.

## 3. Source Code

### A. Fitur Login

Fitur ini digunakan untuk login pengguna dan jika salah memasukkan username atau password sebanyak 3 kali maka program akan berhenti.

**Source Code:**

```

bool login(string UnBenar, string PwBenar) {
    string usernamelogin, passwordlogin;
    int percobaan = 3;
    while (percobaan > 0) {
        cout << "Masukkan Username: ";
        cin >> usernamelogin;
        cout << "Masukkan Password: ";
        cin >> passwordlogin;
        if (usernamelogin == UnBenar && passwordlogin == PwBenar) {
            cout << "Login berhasil" << endl;
            return true;
        } else {
            percobaan--;
            if (percobaan > 0)
  
```

```

        cout << "Username Atau Password Salah, Silahkan Coba Lagi" <<
endl;
        else {
            cout << "Anda Telah Mencoba Login Sebanyak 3 Kali, Program Akan
Dihentikan" << endl;
            return false;
        }
    }
}
return false;
}

```

## B. Fitur Read

Fitur ini digunakan untuk menampilkan semua data yang tersimpan.

### Source Code:

```

void tampilkanData(Character karakter) {
    if (karakter.elemen == 0) {
        cout << "Belum Ada Data Yang Ditambahkan" << endl;
    } else {
        cout << "Daftar Karakter" << endl;
        for (int i = 0; i < karakter.elemen; i++) {
            cout << i + 1 << ". Nama: " << karakter.nama[i]
                << "\n   Bounty: " << karakter.bounty[i] << endl;
        }
    }
}

```

## D. Fitur Create

Fitur ini digunakan untuk menambahkan data baru.

### Source Code:

```

void tambahData(Character& karakter) {
    if (karakter.elemen < MAXChar) {
        cout << "Tambahkan Nama Karakter: ";
        cin >> karakter.nama[karakter.elemen];
        cout << "Berikan Bounty Karakter(Masukkan Dalam Bentuk Angka!): ";
        cin >> karakter.bounty[karakter.elemen];
        karakter.elemen++;
        cout << "Karakter Berhasil Ditambahkan" << endl;
    } else {
        cout << "Kapasitas Penuh! Tidak Dapat Menambah Data." << endl;
    }
}

```

## E. Fitur Update

Fitur ini digunakan untuk mengubah data lama menjadi yang diinginkan.

### Source Code:

```
void ubahData(Character& karakter) {
    if (karakter.elemen == 0) {
        cout << "Belum Ada Data Yang Ditambahkan." << endl;
        return;
    }
    tampilkanData(karakter);
    int index;
    cout << "Masukkan Nomor Data Yang Ingin Diubah: ";
    cin >> index;
    if (index > 0 && index <= karakter.elemen) {
        cout << "Masukkan Nama Baru: ";
        cin >> karakter.nama[index - 1];
        cout << "Masukkan Bounty Baru(Masukkan Dalam Bentuk Angka!): ";
        cin >> karakter.bounty[index - 1];
        cout << "Data Telah Berhasil Dirubah" << endl;
    } else {
        cout << "Nomor Data Tidak Valid" << endl;
    }
}
```

### F. Fitur Delete

Fitur ini digunakan untuk menghapus data yang dipilih.

### Source Code:

```
void ubahData(Character& karakter) {
    if (karakter.elemen == 0) {
        cout << "Belum Ada Data Yang Ditambahkan." << endl;
        return;
    }
    tampilkanData(karakter);
    int index;
    cout << "Masukkan Nomor Data Yang Ingin Diubah: ";
    cin >> index;
    if (index > 0 && index <= karakter.elemen) {
        cout << "Masukkan Nama Baru: ";
        cin >> karakter.nama[index - 1];
        cout << "Masukkan Bounty Baru(Masukkan Dalam Bentuk Angka!): ";
        cin >> karakter.bounty[index - 1];
        cout << "Data Telah Berhasil Dirubah" << endl;
    } else {
        cout << "Nomor Data Tidak Valid" << endl;
    }
}
```

### G. Menu Utama

Ini adalah menu utama dari program.

```

int main() {
    string username = "Fariz";
    string password = "032";
    Character karakter;
    int pilihan;

    cout << "Selamat Datang Di One Piece Pirate List\nSilahkan Login Terlebih
Dahulu\n";

    if (!login(username, password)) {
        return 0;
    }
    do {
        cout << R"(
=====
Pilihan Menu :
1. Tampilkan Semua Data
2. Tambahkan Data
3. Ubah Data
4. Hapus Data
5. Keluar Program
=====
)" << endl;
        cout << "Masukkan Pilihan Anda: ";
        cin >> pilihan;
        switch (pilihan) {
            case 1: tampilkanData(karakter); break;
            case 2: tambahData(karakter); break;
            case 3: ubahData(karakter); break;
            case 4: hapusData(karakter); break;
            case 5:
                cout << "Terimakasih Telah Menggunakan One Piece Pirate List.
Program Akan Dihentikan." << endl;
                break;
            default:
                cout << "Pilihan Tidak Valid" << endl;
                break;
        }
    } while (pilihan != 5);
    return 0;
}

```

#### 4. Hasil Output

```
Selamat Datang Di One Piece Pirate List
Silahkan Login Terlebih Dahulu
Masukkan Username: Fariz
Masukkan Password: 032
Login berhasil

=====
Pilihan Menu :
1. Tampilkan Semua Data
2. Tambahkan Data
3. Ubah Data
4. Hapus Data
5. Keluar Program
=====

Masukkan Pilihan Anda: █
```

Gambar 4.1 Login Berhasil

```
Tambahkan Nama Karakter: Chopper
Berikan Bounty Karakter(Masukkan Dalam Bentuk Angka!): 50
Karakter Berhasil Ditambahkan
```

Gambar 4.2 Menambahkan Data

```
Daftar Karakter
1. Nama: Chopper
   Bounty: 50
```

Gambar 4.3 Menampilkan Data

```
Masukkan Pilihan Anda: 3
1. Nama: Chopper
   Bounty: 50
Masukkan Nomor Data Yang Ingin Diubah: 1
Masukkan Nama Baru: Usopp
Masukkan Bounty Baru(Masukkan Dalam Bentuk Angka!): 500000000
Data Telah Berhasil Dirubah
```

Gambar 4.4 Mengubah Data

```
Masukkan Pilihan Anda: 4
1. Nama: Usopp
   Bounty: 500000000
Masukkan Nomor Data Yang Ingin Dihapus: 1
Data Telah Dihapus
```

Gambar 4.5 Menghapus Data

## 5. Langkah-Langkah Git pada VSCode

```
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test> git add .
warning: in the working copy of 'post-test/post-test-apl-3/2409106032-FarizMuwaffaq-PT-3.cpp', LF will be replaced by CRLF the next time Git
touches it
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test> git add .
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test> git commit -m "Code Posttest 4"
[main fb56d7f] Code Posttest 4
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test> git push
Enumerating objects: 52, done.
Counting objects: 100% (49/49), done.
Delta compression using up to 12 threads
Compressing objects: 100% (37/37), done.
Writing objects: 100% (38/38), 1.16 MiB | 344.00 KiB/s, done.
Total 38 (delta 15), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (15/15), completed with 5 local objects.
To https://github.com/farizmuwaffaq/praktikum-apl.git
 38deb1f..fb56d7f  main -> main
```

Gambar 5.1 Git

### A. Git Add

Digunakan untuk menambahkan file yang akan di commit.

### B. Git Commit

Digunakan untuk menyiapkan file yang akan di commit.

### C. Git Push

Digunakan untuk mengupload file yang telah disiapkan ke GitHub.