

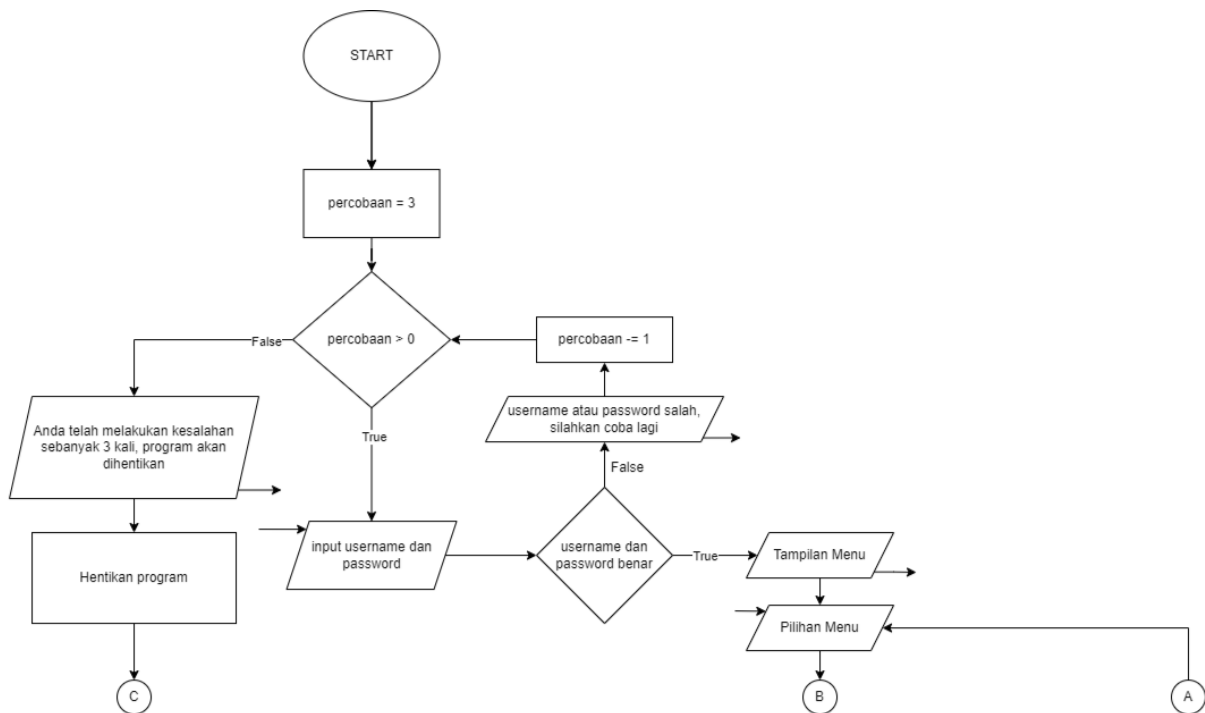
LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN LANJUT



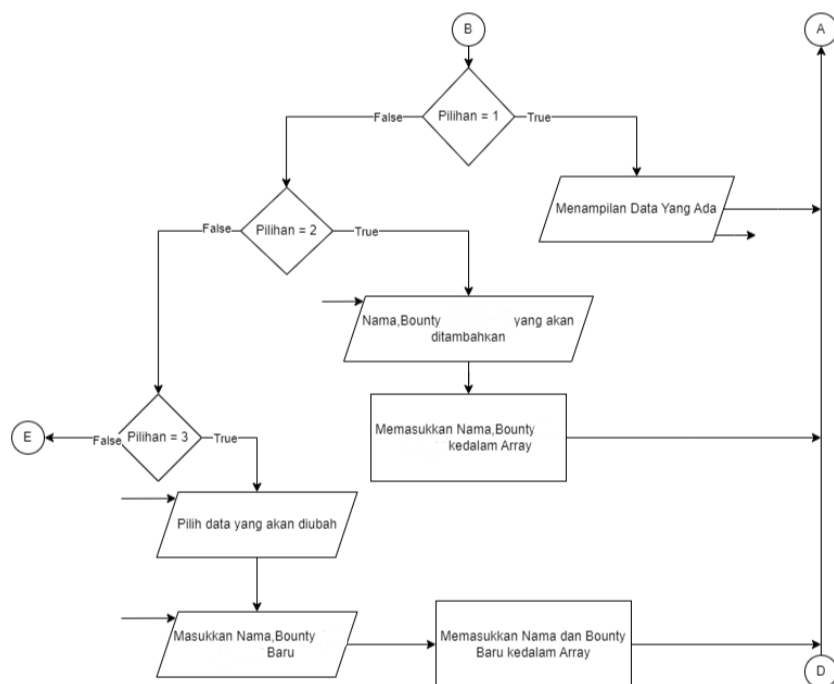
Disusun oleh:
Fariz Muwaffaq (2409106032)
Kelas (A2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

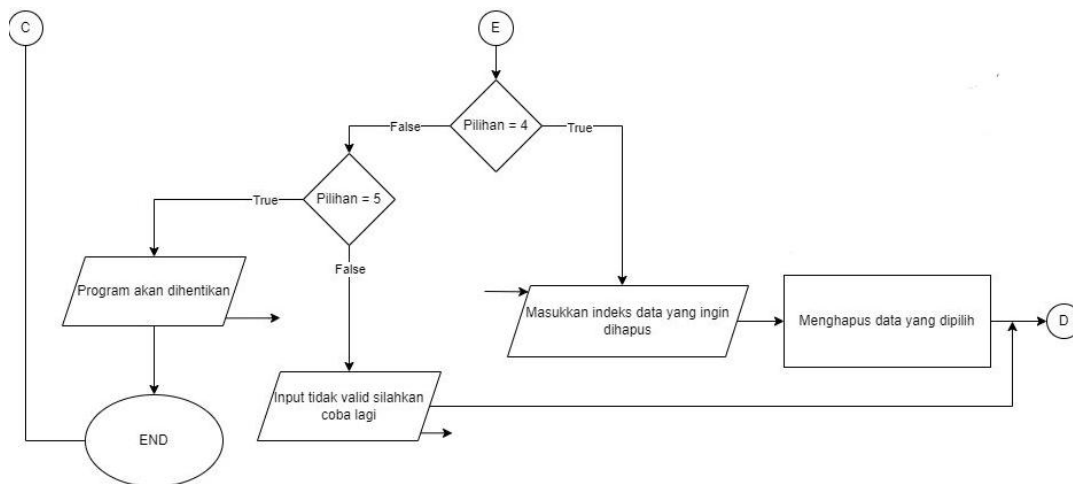
1. Flowchart



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 data baru 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. Bubble Sort

Fungsi ini digunakan untuk mengurutkan nama secara descending dengan menggunakan algoritma bubble sort.

Source Code :

```
void bubbleSortDesc(int bounty[], string nama[], int panjang)
{
    int n = panjang;
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < (n - i - 1); j++)
        {
            if (nama[j] < nama[j + 1])
            {
                string tempNama = nama[j];
                nama[j] = nama[j + 1];
                nama[j + 1] = tempNama;

                int tempBounty = bounty[j];
                bounty[j] = bounty[j + 1];
                bounty[j + 1] = tempBounty;
            }
        }
    }
    cout << "Data Berhasil Di Urutkan" << endl;
}
```

H. Selection Sort

Fungsi ini digunakan untuk mengurutkan bounty secara ascending dengan menggunakan algoritma selection sort.

Source Code :

```
void selectionSortAsc(int bounty[], string nama[], int panjang)
{
    for (int i = 0; i < panjang - 1; i++)
    {
        int min = i;
        for (int j = i + 1; j < panjang; j++)
        {
            if (bounty[j] < bounty[min])
            {
                min = j;
            }
        }
        int tempBounty = bounty[i];
        bounty[i] = bounty[min];
        bounty[min] = tempBounty;

        string tempNama = nama[i];
        nama[i] = nama[min];
    }
}
```

```

        nama[min] = tempNama;
    }
    cout << "Data Berhasil Di Urutkan" << endl;
}

```

I. Insertion Sort

Fungsi ini digunakan untuk mengurutkan bounty secara descending dengan menggunakan algoritma insertion sort.

Source Code :

```

void insertionSortDesc(int bounty[], string nama[], int panjang)
{
    for (int i = 1; i < panjang; i++)
    {
        int keyBounty = bounty[i];
        string keyNama = nama[i];
        int j = i - 1;

        while (j >= 0 && bounty[j] < keyBounty)
        {
            bounty[j + 1] = bounty[j];
            nama[j + 1] = nama[j];
            j--;
        }

        bounty[j + 1] = keyBounty;
        nama[j + 1] = keyNama;
    }
    cout << "Data Berhasil Di Urutkan" << endl;
}

```

J. Menu Utama

Ini adalah menu utama dari program.

Source Code :

```

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. Urutkan Data
6. 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:
            if (karakter.elemen == 0)
            {
                cout << "Belum Ada Data Yang Ditambahkan" << endl;
                break;
            }
            int PilihanSort;
            cout << R"(
=====
Pilihan Sorting :
1. Urutkan Nama Descending
2. Urutkan Bounty Ascending
3. Urutkan Bounty Descending
4. Kembali Ke Menu Utama
=====
)";

            cout << "Masukkan Pilihan Anda: ";
            cin >> PilihanSort;
            switch (PilihanSort)
            {
                case 1:
                    bubbleSortDesc(karakter.bounty, karakter.nama,
karakter.elemen);

                    break;
                case 2:
                    selectionSortAsc(karakter.bounty, karakter.nama,
karakter.elemen);

                    break;
                case 3:
                    insertionSortDesc(karakter.bounty, karakter.nama,
karakter.elemen);

                    break;
                default:
                    cout << "Pilihan Tidak Valid" << endl;

```



```

        break;
    }
    break;
case 6:
    cout << "Terimakasih Telah Menggunakan One Piece Pirate List.
Program Akan Dihentikan." << endl;
    break;
default:
    cout << "Pilihan Tidak Valid" << endl;
    break;
}
} while (pilihan != 6);
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

```

Masukkan Pilihan Anda: 2
Tambahkan Nama Karakter: Eudora
Berikan Bounty Karakter(Masukkan Dalam Bentuk Angka!): 2000
Karakter Berhasil Ditambahkan

```

Gambar 4.2 Menambahkan Data

```
Masukkan Pilihan Anda: 1
Daftar Karakter
1. Nama: Eudora
   Bounty: 2000
2. Nama: Saber
   Bounty: 6500
3. Nama: Bruno
   Bounty: 15000
4. Nama: Fanny
   Bounty: 24000
5. Nama: Alpha
   Bounty: 32000
```

Gambar 4.3 Menampilkan Data

```
Masukkan Pilihan Anda: 3
Daftar Karakter
1. Nama: Eudora
   Bounty: 2000
2. Nama: Saber
   Bounty: 6500
3. Nama: Bruno
   Bounty: 15000
4. Nama: Fanny
   Bounty: 24000
5. Nama: Alpha
   Bounty: 32000
Masukkan Nomor Data Yang Ingin Diubah: 1
Masukkan Nama Baru: Miya
Masukkan Bounty Baru(Masukkan Dalam Bentuk Angka!): 1000
Data Telah Berhasil Dirubah
```

Gambar 4.4 Mengubah Data

```
Masukkan Pilihan Anda: 1
Daftar Karakter
1. Nama: Miya
   Bounty: 1000
2. Nama: Saber
   Bounty: 6500
3. Nama: Bruno
   Bounty: 15000
4. Nama: Fanny
   Bounty: 24000
5. Nama: Alpha
   Bounty: 32000
```

Gambar 4.5 Menampilkan Data Setelah Diubah

```
Masukkan Pilihan Anda: 1
Daftar Karakter
1. Nama: Saber
   Bounty: 6500
2. Nama: Miya
   Bounty: 1000
3. Nama: Fanny
   Bounty: 24000
4. Nama: Bruno
   Bounty: 15000
5. Nama: Alpha
   Bounty: 32000
```

Gambar 4.6 Menampilkan Data Setelah Di Urutkan Menggunakan Bubble Sort

```
Masukkan Pilihan Anda: 1
Daftar Karakter
1. Nama: Miya
   Bounty: 1000
2. Nama: Saber
   Bounty: 6500
3. Nama: Bruno
   Bounty: 15000
4. Nama: Fanny
   Bounty: 24000
5. Nama: Alpha
   Bounty: 32000
```

Gambar 4.7 Menampilkan Data Setelah Di Urutkan Menggunakan Selection Sort

```
Masukkan Pilihan Anda: 1
Daftar Karakter
1. Nama: Alpha
   Bounty: 32000
2. Nama: Fanny
   Bounty: 24000
3. Nama: Bruno
   Bounty: 15000
4. Nama: Saber
   Bounty: 6500
5. Nama: Miya
   Bounty: 1000
```

Gambar 4.8 Menampilkan Data Setelah Di Urutkan Menggunakan Insertion Sort

5. Langkah-Langkah Git pada VSCode

```
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test\post-test-apl-6> git add .
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test\post-test-apl-6> git commit -m "Posttest 6 Code Finish"
[main f75ebd8] Posttest 6 Code Finish
 2 files changed, 240 insertions(+)
   create mode 100644 post-test/post-test-apl-6/2409106032_FarizMuwaffaq-PT-6.cpp
   create mode 100644 post-test/post-test-apl-6/2409106032_FarizMuwaffaq-PT-6.exe
PS C:\Users\USER\Documents\GitHub\praktikum-apl\post-test\post-test-apl-6> git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 678.41 KiB | 8.93 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/farizmuwaffaq/praktikum-apl.git
   b67452c..f75ebd8  main -> main
```

Gambar 5.1 Git

A. Git Add

Digunakan untuk menambahkan file untuk 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.