

LAPORAN PRAKTIKUM
POSTTEST 4
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:
Muhammad Husein Permadi (2409106051)
Kelas (B1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Soal

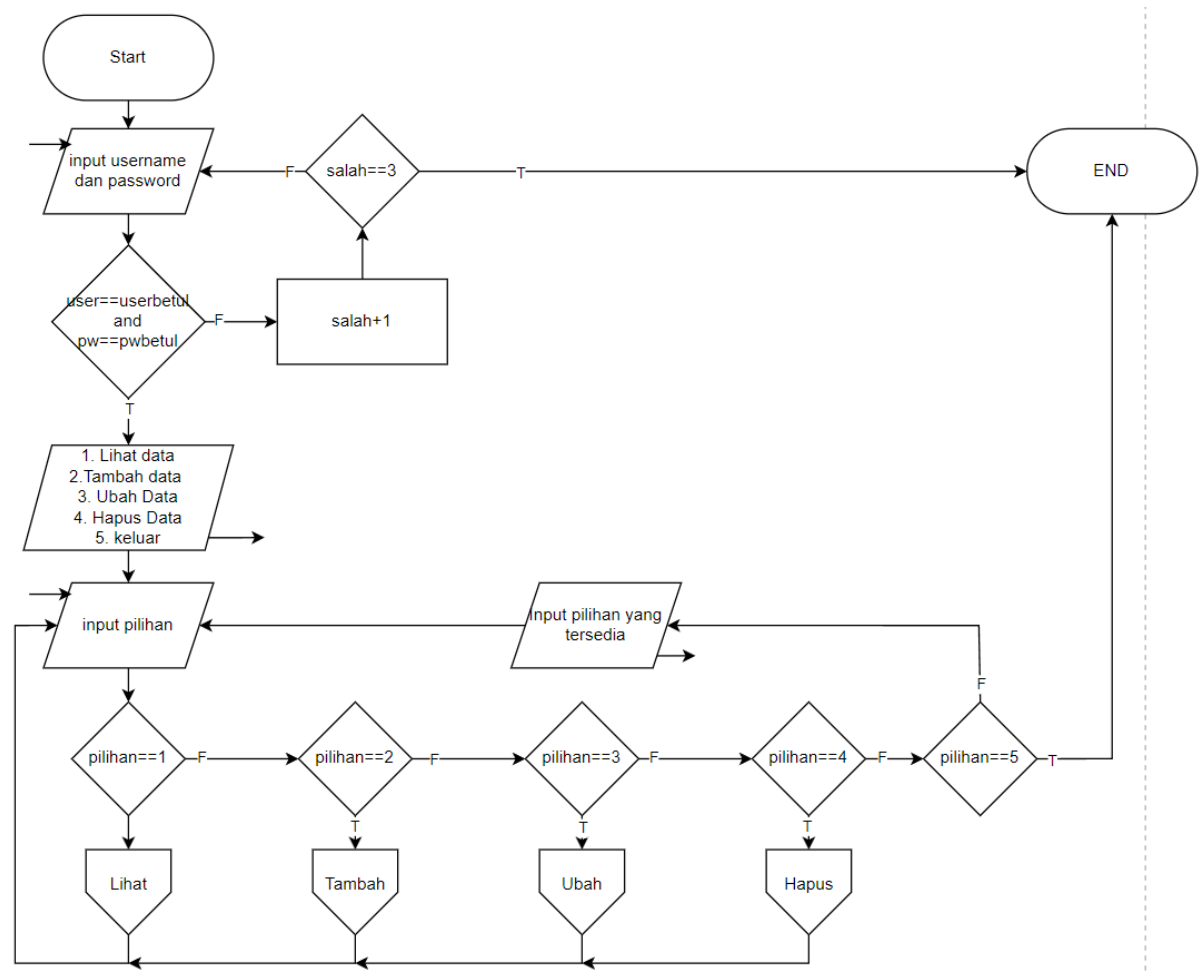
-Dari program kalian sebelumnya pada posttest 4, masing-masing fitur yang ada pada menu dipisahkan dari program utama menjadi sub program

-Fungsi wajib menggunakan parameter

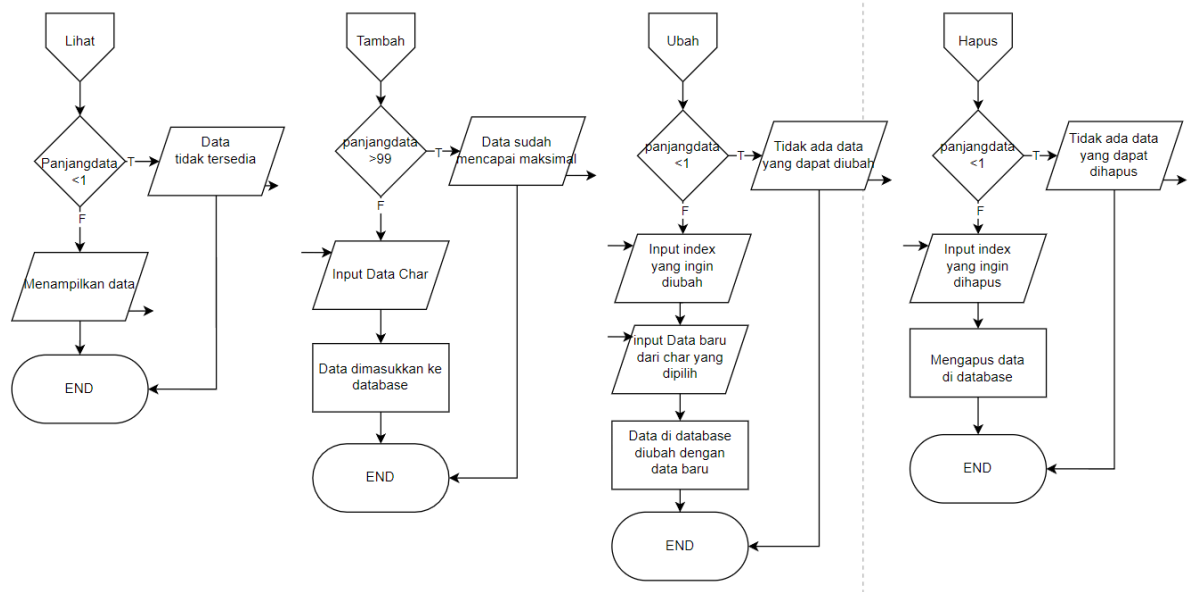
-Wajib menggunakan prosedur

2. Flowchart

Dibawah ini adalah Flowchart dari program Klasifikasi Data karakter game bertema boss raid yang saya buat.



Gambar 1.1 Flowchart Bagian 1



Gambar 1.2 Flowchart Bagian 2

3. Analisis Program

2.1 Deskripsi Singkat Program

Program yang saya buat adalah program klasifikasi data karakter game bertema boss raid yang pada awalnya pengguna diminta untuk login. dan setelah berhasil login maka akan ada 5 menu utama yaitu lihat data, tambah data, ubah data, hapus data dan keluar program. Setelah pengguna memilih keluar maka program akan berakhir.

2.2 Penjelasan Alur & Algoritma

alur dari program yang saya buat pada awalnya user diminta untuk login menggunakan pin, dan jika pin salah 3 kali maka akan otomatis keluar program, dan jika pin benar maka akan masuk ke program Data Karakter. di dalam program Data karakter terdapat 5 menu utama yaitu lihat data, tambah data, ubah data, hapus data, dan keluar.

Pada menu lihat data, user akan ditampilkan seluruh data karakter yang sudah ditambahkan saat ini, mulai dari nama, role, weapon, power dan pasif karakter. Kemudian pada menu tambah data pengguna menginput data karakter baru yang ingin ditambahkan, jika kapasitas data masih cukup maka data akan ditambahkan, sedangkan jika tidak cukup, maka data tidak akan ditambahkan. Kemudian pada menu ubah data, pengguna akan memilih index yang akan diubah kemudian pengguna akan menginput data baru untuk index yang dipilih, setelah itu program akan mengganti data lama dengan data baru. selanjutnya jika pengguna memilih hapus data, pengguna akan memilih index dari data yang ingin dihapus, setelah itu program akan menghapus data dari index yang dipilih. Kemudian jika pengguna memilih menu keluar maka program akan berhenti.

Algoritma Data Karakter

1. Masukkan username dan password.
2. Jika username dan password benar maka masuk ke program utama.
3. Jika username atau password salah 3 kali maka program berhenti.
4. Masukkan Pilihan antara lihat data, tambah data, ubah data, hapus data dan keluar.
5. Jika memilih lihat data maka akan menampilkan data saat ini dan kembali ke menu utama.

6. Jika memilih tambah data maka program akan mengecek apakah data lebih dari 99..
7. Jika jumlah data lebih dari 99 maka penambahan data akan gagal dan kembali ke menu utama
8. Jika jumlah data kurang dari atau sama dengan 99 maka pengguna akan menginput data karakter baru, setelah itu program akan menambahkan data karakter baru ke database dan kembali ke menu utama.
9. Jika memilih ubah data maka masukkan index yang ingin di ubah kemudian masukkan data baru untuk mengubah index. setelah itu program akan mengganti data lama dengan data baru, dan kembali ke menu utama.
10. Jika memilih hapus data maka pengguna akan menginput index yang ingin dihapus
11. Program akan menghapus index yang di input pengguna, dan pengguna akan kembali ke menu utama.
12. Jika memilih Keluar maka program akan berakhir.

4. Source Code dan penjelasan code program

A. Perubahan Database

Program dibawah ini menampilkan database baru menggunakan array of struct yang sebelumnya menggunakan array saja.

```
struct infochara {  
    string role;  
    string pasif;  
    string weapon;  
    string power;  
};  
  
struct chara {  
    string nama;  
    struct infochara info;  
};  
  
struct chara chr[max];
```

```
chr[0].nama = "Gus Akira Nakai 99";  
chr[0].info.role = "DPS Warrior";  
chr[0].info.pasif = "Menyerang akan mengurangi Defense musuh sebanyak 5%  
hingga maksimal 40%";  
chr[0].info.weapon = "ShogunXmodification";  
chr[0].info.power = "The Chosen One";  
  
chr[1].nama = "Miftah JRM";  
chr[1].info.role = "Mage Debuffer";  
chr[1].info.pasif = "Menghina semua musuh,membuat stat dasar musuh berkurang  
20%";  
chr[1].info.weapon = "Tea of Sadness";  
chr[1].info.power = "Psychology Damage";  
  
chr[2].nama = "Tyrant King, The Mulkowi";  
chr[2].info.role = "Buff Support";  
chr[2].info.pasif = "Memerintahkan Sekutu untuk menyerang, meningkatkan 45%  
damage kepada musuh";  
chr[2].info.weapon = "Ring Of Absolute Order";  
chr[2].info.power = "Voice Order";
```

B. Program Login

Program di bawah digunakan untuk membuat fitur login yang dimana jika pengguna salah memasukkan username atau password 3 kali maka akan keluar dari program.

```
cout<<endl<<"Database Character BossRaid"<<endl<<endl;
```

```

for(int kesempatan=3;kesempatan>0;kesempatan--){
    cout<<"Masukkan Username dan Password !!!"<<endl;
    cout<<"Masukkan Username : ";
    cin>>user;
    cout<<endl<<"Masukkan Password : ";
    cin>>password;
    if(user==userbetul and password==pwbetul){
        login='y';
        cout<<"Berhasil masuk !!!"<<endl;
        break;
    }
    else{
        cout<<"Username atau Password yang anda masukkan salah"<<endl;
        salah=salah+1;
    }
}
if(salah==3){
    cout<<"Anda mencapai batas salah !"<<endl;
    cout<<"Keluar dari program !";
}

```

Pada program di atas akan mengecek apakah username dan password yang dimasukkan pengguna benar (`user==userbetul and password==pwbetul`) atau salah (`else`). dan jika pin sudah benar maka akan langsung menghentikan perulangan dengan (`break`;) dan masuk ke program utama.

Namun jika user menginputkan pin yang salah, maka program akan menghitung jumlah kesalahan (`salah=salah+1`), dan jika memasukkan salah 3 kali maka program akan langsung berhenti.

C. Program menampilkan Menu utama

Program di bawah ini digunakan untuk fitur menampilkan menu utama dan memasukkan pilihan.

```

while (login=='y'){
    cout<<endl<<"Database Character"<<endl<<endl;
    cout<<"Pilih menu"<<endl;
    cout<<"1. Lihat Database"<<endl;
    cout<<"2. Tambahkan Informasi Character"<<endl;
    cout<<"3. Ubah Informasi Character"<<endl;
    cout<<"4. Hapus Informatika Character"<<endl;
    cout<<"5. Keluar Program"<<endl<<endl;
    int pilih;
    cout<<"Masukkan Pilihan : ";
    cin>> pilih;
}

```

Pada program di atas akan menampilkan menu utama program yang menggunakan perulangan(`while (login=='y')`) agar user bisa memilih beberapa pilihan tanpa mengulang program dan keluar program.

D. Program pilihan menampilkan data

Program di bawah ini digunakan untuk menampilkan pilihan pertama yaitu menampilkan data.

```
void lihatData(chara chr[], int panjangdata) {
    if (panjangdata > 0) {
        for (int i = 0; i < panjangdata; i++) {
            cout << i + 1 << ". Nama      : " << chr[i].nama << endl;
            cout << "      Role      : " << chr[i].info.role << endl;
            cout << "      Weapon    : " << chr[i].info.weapon << endl;
            cout << "      Power     : " << chr[i].info.power << endl;
            cout << "      Passive   : " << chr[i].info.pasif << endl << endl;
        }
    } else {
        cout << "Tidak ada data untuk ditampilkan" << endl;
    }
}
```

Pada Program di atas jika user memilih pilihan pertama (`if(pilih==1)`), yaitu menampilkan data, maka akan memberikan output untuk menampilkan data. dan di atas menggunakan perulangan(`for(int nodata=0;panjangdata>0;panjangdata--,no++,nodata++)`) untuk menampilkan seluruh data dalam array agar dapat menampilkan data yang telah diubah atau ditambahkan.

E. Program pilihan tambah

Program di bawah ini digunakan untuk fitur menambahkan data.

```
void tambahData(chara chr[], int& panjangdata) {
    if (panjangdata < max) {
        cin.ignore();
        cout << "Masukkan Nama Char baru   : ";
        getline(cin, chr[panjangdata].nama);
        cout << "Masukkan Role Char baru   : ";
        getline(cin, chr[panjangdata].info.role);
        cout << "Masukkan Weapon char baru : ";
        getline(cin, chr[panjangdata].info.weapon);
        cout << "Masukkan Power char baru  : ";
        getline(cin, chr[panjangdata].info.power);
        cout << "Masukkan Pasif Char baru  : ";
        getline(cin, chr[panjangdata].info.pasif);
        panjangdata++;
    } else {
        cout << "Panjang Data sudah mencapai maksimal !!!" << endl;
    }
}
```

Pada program di atas jika pengguna memilih pilihan kedua (`else if(pilih==2)`), yaitu menambahkan data, maka program akan meminta user untuk menginputkan data karakter baru contohnya nama(`cout<<"Masukkan Nama Char baru:"; cin.ignore(); getline(cin, chr[panjangdata].nama);`).

Namun sebelumnya program akan mengecek apakah database masih cukup untuk menambah data (`if(panjangdata<100)`), jika tidak maka data tidak akan ditambah.

F. Program pilihan mengubah data

Program di bawah ini digunakan untuk fitur mengubah data.

```
void ubahData(chara chr[], int panjangdata) {
    if (panjangdata < 1) {
        cout << "Tidak ada data untuk diubah" << endl;
    } else {
        for (int i = 0; i < panjangdata; i++) {
            cout << i + 1 << ". Nama      : " << chr[i].nama << endl;
        }
        int pilihdata;
        cout << "Pilih data yang ingin diubah : ";
        cin >> pilihdata;
        if (pilihdata < 1 || pilihdata > panjangdata) {
            cout << "Data tidak ditemukan atau tidak valid!" << endl;
        } else {
            cin.ignore();
            cout << "Mengubah data untuk " << chr[pilihdata - 1].nama <<
endl;

            cout << "Masukkan Nama Char baru   : ";
            getline(cin, chr[pilihdata - 1].nama);
            cout << "Masukkan Role Char baru   : ";
            getline(cin, chr[pilihdata - 1].info.role);
            cout << "Masukkan Weapon char baru : ";
            getline(cin, chr[pilihdata - 1].info.weapon);
            cout << "Masukkan Power char baru  : ";
            getline(cin, chr[pilihdata - 1].info.power);
            cout << "Masukkan Pasif Char baru  : ";
            getline(cin, chr[pilihdata - 1].info.pasif);
        }
    }
}
```

Pada program di atas jika pengguna memilih pilihan ketiga yaitu pilihan mengubah data maka program akan meminta user untuk memasukkan index yang ingin diubah(`cout<<"Pilih data yang ingin diubah : "; cin>>pilihdata;`), kemudian program akan mengecek apakah index yang dimasukkan benar atau salah, jika index ditemukan(benar) maka pengguna akan diminta untuk menginput data baru yang digunakan untuk mengubah data lama, setelah itu data lama akan diubah ke data baru.

G. Program pilihan hapus data

Program dibawah ini adalah untuk fitur hapus data

```
void hapusData(chara chr[], int& panjangdata) {
    if (panjangdata < 1) {
```

```

    cout << "Tidak ada data untuk dihapus" << endl;
} else {
    for (int i = 0; i < panjangdata; i++) {
        cout << i + 1 << ". Nama      : " << chr[i].nama << endl;
    }
    int indexCH;
    cout << "Pilih data nomor berapa yang ingin dihapus : ";
    cin >> indexCH;
    if (indexCH < 1 || indexCH > panjangdata) {
        cout << "Data tidak ditemukan" << endl;
    } else {
        for (int i = indexCH - 1; i < panjangdata - 1; i++) {
            chr[i] = chr[i + 1];
        }
        panjangdata--;
        cout << "Berhasil menghapus Data" << endl;
    }
}

```

Pada program diatas adalah jika pengguna memilih pilihan untuk menghapus data, pengguna akan diminta untuk menginput index yang ingin dihapus(`cout<<endl<<"Pilih data nomor berapa yang ingin dihapus: "; cin>>indexCH;`). kemudian program akan menghapus element dari index tersebut dan semua element dibelakangnya akan digeser kedepan untuk memenuhi index dari awal.

H. Program pilihan keluar dan input diluar pilihan

Program di bawah ini digunakan untuk fitur keluar dari perulangan dan mengakhiri program, dan jika user menginput masukan diluar pilihan.

```

else if(pilih==5){
    cout<<"Keluar dari program";
    break;
}

```

Program di atas jika pengguna memilih pilihan 5 (`else if(pilih==5)`), yaitu pilihan keluar dari program. Maka program akan langsung menghentikan perulangan dengan `break(break;)` dan program akan berakhir karena tidak ada program lagi yang dijalankan.

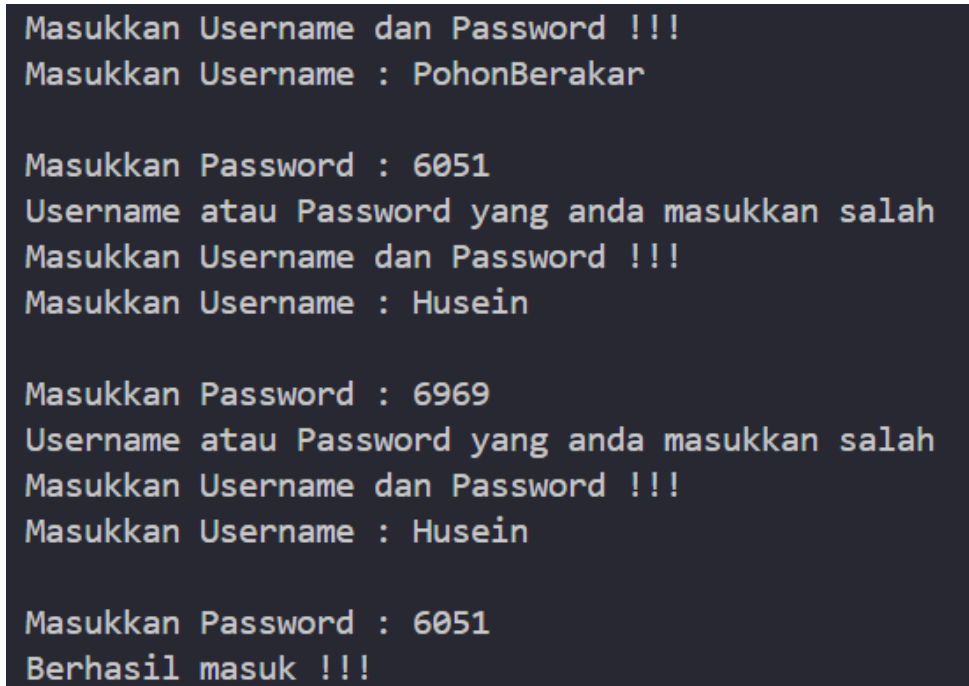
4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. Uji coba gagal login
Jika user gagal memasukkan username dan password yang benar 3 kali maka akan keluar program.
2. Jika user menginput index untuk diubah atau dihapus, maka program akan menampilkan output yang memberitahu bahwa index tidak tersedia, kemudian pengguna akan kembali ke menu utama.

4.2 Hasil Output

A. Output Login



```
Masukkan Username dan Password !!!  
Masukkan Username : PohonBerakar  
  
Masukkan Password : 6051  
Username atau Password yang anda masukkan salah  
Masukkan Username dan Password !!!  
Masukkan Username : Husein  
  
Masukkan Password : 6969  
Username atau Password yang anda masukkan salah  
Masukkan Username dan Password !!!  
Masukkan Username : Husein  
  
Masukkan Password : 6051  
Berhasil masuk !!!
```

Gambar 4.1 Output Login

B. Output menu utama dan lihat data

```
Database Character

Pilih menu
1. Lihat Database
2. Tambahkan Informasi Character
3. Ubah Informasi Character
4. Hapus Informatika Character
5. Keluar Program

Masukkan Pilihan : 1
1. Nama      : Gus Akira Nakai 99
   Role      : DPS Warrior
   Weapon     : ShogunXmodification
   Power      : The Chosen One
   Passive    : Menyerang akan mengurangi Defense musuh sebanyak 5% hingga maksimal 40%

2. Nama      : Miftah JRM
   Role      : Mage Debuffer
   Weapon     : Tea of Sadness
   Power      : Psychology Damage
   Passive    : Menghina semua musuh,membuat stat dasar musuh berkurang 20%

3. Nama      : Tyrant King, The Mulkowi
   Role      : Buff Support
   Weapon     : Ring Of Absolute Order
   Power      : Voice Order
   Passive    : Memerintahkan Sekutu untuk menyerang, meningkatkan 45% damage kepada musuh
```

Gambar 4.2 Output menu utama dan lihat data

C. Output tambah data

```
Database Character

Pilih menu
1. Lihat Database
2. Tambahkan Informasi Character
3. Ubah Informasi Character
4. Hapus Informatika Character
5. Keluar Program

Masukkan Pilihan : 2
Masukkan Nama Char baru   : Budiono Siregar
Masukkan Role Char baru   : Kapal Lawd
Masukkan Weapon char baru : Meriam
Masukkan Power char baru  : Batubara
Masukkan Pasif Char baru  : Saat bergerak di air,maka movement speed +400%
```

Gambar 4.3 Output tambah data

D. Output Ubah data

```
Database Character

Pilih menu
1. Lihat Database
2. Tambahkan Informasi Character
3. Ubah Informasi Character
4. Hapus Informatika Character
5. Keluar Program

Masukkan Pilihan : 3
1. Nama      : Gus Akira Nakai 99
2. Nama      : Miftah JRM
3. Nama      : Tyrant King, The Mulkowi
4. Nama      : Budiono Siregar
Pilih data yang ingin diubah : 4
Mengubah data untuk Budiono Siregar
Masukkan Nama Char baru   : King Budi
Masukkan Role Char baru   : Ranged DPS
Masukkan Weapon char baru : Crystal Bow
Masukkan Power char baru  : Ice Elementalist
Masukkan Pasif Char baru  : Menyerang memberikan ice shard
```

Gambar 4.4 Output Ubah Data

E. Output Hapus data

Database Character

Pilih menu

1. Lihat Database
2. Tambahkan Informasi Character
3. Ubah Informasi Character
4. Hapus Informatika Character
5. Keluar Program

Masukkan Pilihan : 4

1. Nama : Gus Akira Nakai 99
2. Nama : Miftah JRM
3. Nama : Tyrant King, The Mulkowi
4. Nama : King Budi

Pilih data nomor berapa yang ingin dihapus : 2
Berhasil menghapus Data

4.5 Output Hapus data

F. Output Keluar Program

Database Character

Pilih menu

1. Lihat Database
2. Tambahkan Informasi Character
3. Ubah Informasi Character
4. Hapus Informatika Character
5. Keluar Program

Masukkan Pilihan : 5

Keluar dari program

PS D:\TugasSMT2\Praktikum_APL\Post-test\PostTest_APL_2>

Gambar 4.6 Output Keluar Program

5. Langkah - langkah Git

A. Git add

Perintah git add digunakan untuk menambahkan file apa saja sebelum dilakukannya commit.

```
PS D:\TugasSMT2\Praktikum_APL\Post-test\PostTest_APL_3> git add .
```

Gambar 5.1 Git add

B. Git commit

Perintah git commit digunakan untuk menyimpan perubahan yang telah di tambahkan ke dalam repository git

```
PS D:\TugasSMT2\Praktikum_APL> git commit -m "Finish Post Test 4"
[main 138b234] Finish Post Test 4
3 files changed, 178 insertions(+)
create mode 100644 Kelas/Pertemuan-4/fugbsi.cpp
create mode 100644 Post-test/PostTest_APL_4/2409106051_MuhammadHuseinPermadi_PostTest_4.cpp
create mode 100644 Post-test/PostTest_APL_4/2409106051_MuhammadHuseinPermadi_PostTest_4.exe
```

Gambar 5.2 Git Commit

C. Git Push

git push digunakan untuk mengunggah perubahan dari repository lokal ke github

```
PS D:\TugasSMT2\Praktikum_APL> git push origin main
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 16 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (9/9), 678.73 KiB | 3.45 MiB/s, done.
Total 9 (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/PohonBerakar/Praktikum-APL.git
    091eeb7..138b234  main -> main
PS D:\TugasSMT2\Praktikum_APL>
```

Gambar 5.3 Git Push