

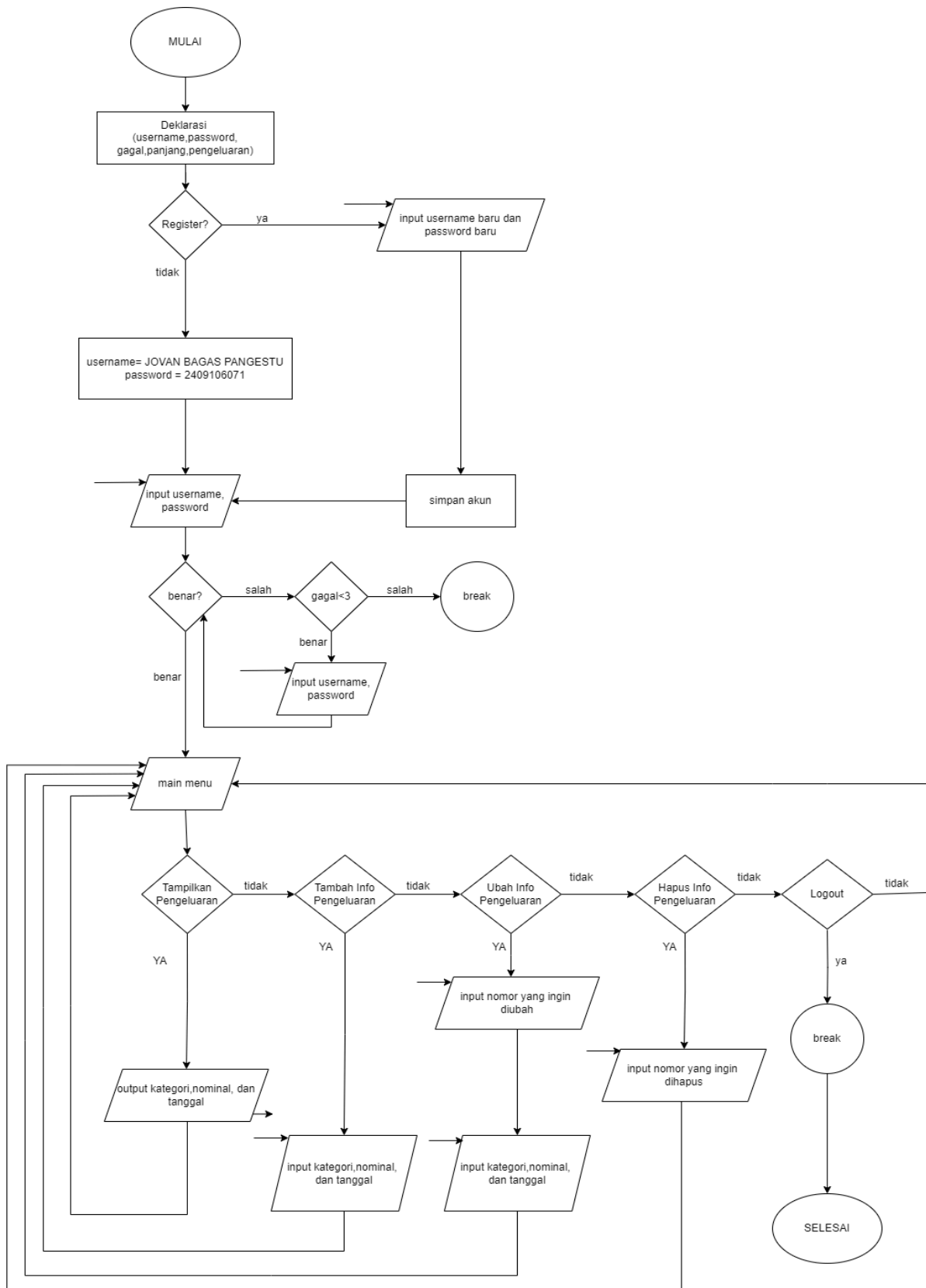
LAPORAN PRAKTIKUM
POSTTEST (3)
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:
Jovan Bagas Pangestu (2409106071)
Kelas (B2'24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



2. Analisis Program

Program ini bertujuan untuk membantu pengguna dalam mencatat dan mengelola pengeluaran mereka secara lebih terstruktur dan efisien. Dengan adanya fitur login dan registrasi, setiap pengguna dapat memiliki akun pribadi yang menyimpan data pengeluaran mereka dengan aman. Program menggunakan struktur data struct untuk mengorganisir informasi pengguna dan daftar pengeluaran, termasuk kategori, nominal, serta tanggal transaksi, sehingga memudahkan dalam pencatatan dan pemantauan keuangan.

Selain itu, program ini memiliki sistem keamanan yang membatasi jumlah percobaan login yang gagal guna mencegah akses tidak sah. Setelah berhasil login, pengguna dapat mengakses berbagai fitur, seperti melihat daftar pengeluaran, menambah pengeluaran baru, mengedit, dan menghapus data yang sudah ada. Dengan adanya fitur ini, pengguna dapat mengelola keuangan mereka dengan lebih baik, mengetahui pola pengeluaran, serta membuat keputusan finansial yang lebih bijak berdasarkan catatan yang tersimpan dalam program.

3. Source Code

A. Struct Sistem

```
struct Pengeluaran {
    string kategori;
    string nominal;
    string tanggal;
};

struct User {
    string username;
    string password;
    Pengeluaran pengeluaran[100];
    int panjangPengeluaran;
};
```

B. User dan Proses Masuk

```
User user[100];
int jumlahUser = 1;
int max_gagal = 3;

int login(string username, string password) {
    for (int i = 0; i < jumlahUser; i++) {
        if (user[i].username == username && user[i].password == password) {
            return i;
        }
    }
    return -1;
}

int registerUser(string username, string password) {
    user[jumlahUser].username = username;
    user[jumlahUser].password = password;
    user[jumlahUser].panjangPengeluaran = 0;
    jumlahUser++;
    return jumlahUser - 1;
}
```

C. Main Program

```
int main() {
    user[0].username = "JOVAN BAGAS PANGESTU";
    user[0].password = "2409106071";
    user[0].panjangPengeluaran = 0;
```

```

int gagal = 0;
int loginUserIndex = -1;

while (gagal < max_gagal) {
    int pilihan;
    cout << "1. Login" << endl;
    cout << "2. Register" << endl;
    cout << "Pilihan: ";
    cin >> pilihan;

    if (pilihan == 1) {
        string username, password;
        cout << "Masukkan username: ";
        cin.ignore();
        getline(cin, username);
        cout << "Masukkan password: ";
        getline(cin, password);

        loginUserIndex = login(username, password);
        if (loginUserIndex != -1) {
            cout << "Login Berhasil!" << endl;
            cout << "Selamat datang, " << user[loginUserIndex].username
<< endl;

```

D. Tampilan Info Pengeluaran

```

case 1:
    // Menampilkan Info Pengeluaran
    if (user[loginUserIndex].panjangPengeluaran ==
0) {
        cout << "Belum ada data pengeluaran." <<
endl;
    } else {
        cout << "\nDaftar Pengeluaran Bulanan:" <<
endl;

        cout << "No. | Kategori | Nominal |
Tanggal" << endl;

        cout <<
"-----" << endl;
        for (int i = 0; i <
user[loginUserIndex].panjangPengeluaran; i++) {
            cout << i + 1 << " | " <<
user[loginUserIndex].pengeluaran[i].kategori << " | "
<<
user[loginUserIndex].pengeluaran[i].nominal << " | "
<<
user[loginUserIndex].pengeluaran[i].tanggal << endl;

```

```

    }
}
break;

```

E. Tambah Info Pengeluaran

```

case 2:
    // Menambah Info Pengeluaran
    if (user[loginUserIndex].panjangPengeluaran <
100) {
        cout << "Masukkan kategori pengeluaran: ";
        cin.ignore();
        getline(cin,
user[loginUserIndex].pengeluaran[user[loginUserIndex].panjangPengeluaran].ka
tegori);
        cout << "Masukkan nominal pengeluaran: ";
        getline(cin,
user[loginUserIndex].pengeluaran[user[loginUserIndex].panjangPengeluaran].no
minal);
        cout << "Masukkan tanggal pengeluaran
(tgl-bln-tahun): ";
        getline(cin,
user[loginUserIndex].pengeluaran[user[loginUserIndex].panjangPengeluaran].ta
nggal);
        user[loginUserIndex].panjangPengeluaran++;
        cout << "Pengeluaran berhasil ditambahkan!"
<< endl;
    } else {
        cout << "Kapasitas penuh! Tidak bisa
menambah data lagi." << endl;
    }
    break;

```

F. Ubah Data

```

case 3:
    // Ubah Info Pengeluaran
    if (user[loginUserIndex].panjangPengeluaran ==
0) {
        cout << "Belum ada data pengeluaran." <<
endl;
    } else {
        cout << "\nDaftar Pengeluaran Bulanan:" <<
endl;
        cout << "No. | Kategori | Nominal |
Tanggal" << endl;
        cout <<
"-----" << endl;

```

```

        for (int i = 0; i <
user[loginUserIndex].panjangPengeluaran; i++) {
            cout << i + 1 << "    | " <<
user[loginUserIndex].pengeluaran[i].kategori << "    | "
                <<
user[loginUserIndex].pengeluaran[i].nominal << "    | "
                <<
user[loginUserIndex].pengeluaran[i].tanggal << endl;
        }

        int index;
        cout << "Masukkan nomor pengeluaran yang
ingin diubah: ";

        cin >> index;

        if (index > 0 && index <=
user[loginUserIndex].panjangPengeluaran) {
            cout << "Masukkan kategori baru: ";
            cin.ignore();
            getline(cin,
user[loginUserIndex].pengeluaran[index - 1].kategori);
            cout << "Masukkan nominal baru: ";
            getline(cin,
user[loginUserIndex].pengeluaran[index - 1].nominal);
            cout << "Masukkan tanggal baru
(tgl-bln-tahun): ";
            getline(cin,
user[loginUserIndex].pengeluaran[index - 1].tanggal);
            cout << "Pengeluaran berhasil diubah!"
<< endl;
        } else {
            cout << "Nomor tidak valid!" << endl;
        }
    }
    break;
}

```

G. Hapus Data

```

case 4:
    // Hapus Info Pengeluaran
    if (user[loginUserIndex].panjangPengeluaran ==
0) {
        cout << "Belum ada data pengeluaran." <<
endl;
    } else {
        cout << "\nDaftar Pengeluaran Bulanan:" <<
endl;
    }
}

```

```

        cout << "No. | Kategori | Nominal |
Tanggal" << endl;

        cout <<
        "-----" << endl;
        for (int i = 0; i <
user[loginUserIndex].panjangPengeluaran; i++) {
            cout << i + 1 << " | " <<
user[loginUserIndex].pengeluaran[i].kategori << " | "
            <<
user[loginUserIndex].pengeluaran[i].nominal << " | "
            <<
user[loginUserIndex].pengeluaran[i].tanggal << endl;
        }

        int index;
        cout << "Masukkan nomor pengeluaran yang
ingin dihapus: ";

        cin >> index;

        if (index > 0 && index <=
user[loginUserIndex].panjangPengeluaran) {
            for (int i = index - 1; i <
user[loginUserIndex].panjangPengeluaran - 1; i++) {
                user[loginUserIndex].pengeluaran[i]
= user[loginUserIndex].pengeluaran[i + 1];
            }

            user[loginUserIndex].panjangPengeluaran--;
            cout << "Pengeluaran berhasil dihapus!"
<< endl;
        } else {
            cout << "Nomor tidak valid!" << endl;
        }
    }
    break;
}

```

H. Register User

```

} else if (pilihan == 2) {
    string username, password;
    cout << "Masukkan username baru: ";
    cin.ignore();
    getline(cin, username);
    cout << "Masukkan password baru: ";
    getline(cin, password);
    registerUser(username, password);
    cout << "Registrasi berhasil!" << endl;
}

```


4. Hasil Output

```
1. Login
2. Register
Pilihan: 1
Masukkan username: JOVAN BAGAS PANGESTU
Masukkan password: 2409106071
Login Berhasil!
Selamat datang, JOVAN BAGAS PANGESTU
Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 1
Belum ada data pengeluaran.
Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 2
Masukkan kategori pengeluaran: AIR
Masukkan nominal pengeluaran: 25000
Masukkan tanggal pengeluaran (tgl-bln-tahun): 01-01-2001
Pengeluaran berhasil ditambahkan!
Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 1

Daftar Pengeluaran Bulanan:
No. | Kategori   | Nominal   | Tanggal
-----
1   | AIR        | 25000     | 01-01-2001
```

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 2

Masukkan kategori pengeluaran: API

Masukkan nominal pengeluaran: 80000

Masukkan tanggal pengeluaran (tgl-bln-tahun): 02-02-2001

Pengeluaran berhasil ditambahkan!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 2

Masukkan kategori pengeluaran: LISTRIK

Masukkan nominal pengeluaran: 100.000

Masukkan tanggal pengeluaran (tgl-bln-tahun): 02-01-2001

Pengeluaran berhasil ditambahkan!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 2

Masukkan kategori pengeluaran: JAJAN

Masukkan nominal pengeluaran: 200.000

Masukkan tanggal pengeluaran (tgl-bln-tahun): 30-01-2001

Pengeluaran berhasil ditambahkan!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 1

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25000	01-01-2001
2	API	80000	02-02-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 3

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25000	01-01-2001
2	API	80000	02-02-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Masukkan nomor pengeluaran yang ingin diubah: 0

Nomor tidak valid!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 3

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25000	01-01-2001
2	API	80000	02-02-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Masukkan nomor pengeluaran yang ingin diubah: -1

Nomor tidak valid!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 3

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25000	01-01-2001
2	API	80000	02-02-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Masukkan nomor pengeluaran yang ingin diubah: 9

Nomor tidak valid!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 3

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25000	01-01-2001
2	API	80000	02-02-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Masukkan nomor pengeluaran yang ingin diubah: 2

Masukkan kategori baru: API

Masukkan nominal baru: 70.000

Masukkan tanggal baru (tgl-bln-tahun): 13-01-2001

Pengeluaran berhasil diubah!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 3

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25000	01-01-2001
2	API	70.000	13-01-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Masukkan nomor pengeluaran yang ingin diubah: 1

Masukkan kategori baru: AIR

Masukkan nominal baru: 25.000

Masukkan tanggal baru (tgl-bln-tahun): 01-01-2001

Pengeluaran berhasil diubah!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 1

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25.000	01-01-2001
2	API	70.000	13-01-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

Pilihan: 4

Daftar Pengeluaran Bulanan:

No.	Kategori	Nominal	Tanggal
-----	----------	---------	---------

1	AIR	25.000	01-01-2001
2	API	70.000	13-01-2001
3	LISTRIK	100.000	02-01-2001
4	JAJAN	200.000	30-01-2001

Masukkan nomor pengeluaran yang ingin dihapus: 3

Pengeluaran berhasil dihapus!

Menu Program

1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar

```

Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 4

Daftar Pengeluaran Bulanan:
No. | Kategori | Nominal | Tanggal
-----
1 | AIR | 25.000 | 01-01-2001
2 | API | 70.000 | 13-01-2001
3 | JAJAN | 200.000 | 30-01-2001
Masukkan nomor pengeluaran yang ingin dihapus: 0
Nomor tidak valid!
Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 4

Daftar Pengeluaran Bulanan:
No. | Kategori | Nominal | Tanggal
-----
1 | AIR | 25.000 | 01-01-2001
2 | API | 70.000 | 13-01-2001
3 | JAJAN | 200.000 | 30-01-2001
Masukkan nomor pengeluaran yang ingin dihapus: -700
Nomor tidak valid!
Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 5
Logout Berhasil!
PS C:\Users\pjova\Downloads> cd "c:\Users\pjova\Downlo
.\2409106071-JovanBagasPangestu-PT-3 }

```

```
1. Login
2. Register
Pilihan: 2
Masukkan username baru: YUYUN
Masukkan password baru: NUYUY
Registrasi berhasil!
1. Login
2. Register
Pilihan: 1
Masukkan username: YUYUN
Masukkan password: NUYUY
Login Berhasil!
Selamat datang, YUYUN
Menu Program
1. Tampilkan Info Pengeluaran
2. Tambah Info Pengeluaran
3. Ubah Info Pengeluaran
4. Hapus Info Pengeluaran
5. Keluar
Pilihan: 5
Logout Berhasil!
PS C:\Users\niova\Downloads> █
```


5. Langkah-Langkah Git pada VSCode

A. GIT ADD .

Menambahkan file ke staggging area untuk di commit.

```
pjova@JovanBagas MINGW64 ~/Downloads/post-test/post-test-apl-3 (main)
$ git add .
```

B. GIT COMMIT PUSH -M

Menyimpan file dari staggging area ke repository disertai pesan singkat deskriptif

```
pjova@JovanBagas MINGW64 ~/Downloads/post-test/post-test-apl-3 (main)
$ git commit -m "file cpp dan exe pt-3"
[main d214cf7] file cpp dan exe pt-3
2 files changed, 205 insertions(+)
create mode 100644 Downloads/post-test/post-test-apl-3/2409106071-JovanBagasPangestu-PT-3.cpp
create mode 100644 Downloads/post-test/post-test-apl-3/2409106071-JovanBagasPangestu-PT-3.exe
```

C. GIT PUSH ORIGIN MAIN

Mengunggah file dari branch main di repo lokal ke repo remot

```
pjova@JovanBagas MINGW64 ~/Downloads/post-test/post-test-apl-3 (main)
$ git push origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 677.83 KiB | 7.97 MiB/s, done.
Total 7 (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/JovanBagasPangestu/praktikum-apl.git
267f85b..d214cf7 main -> main
```

D. History Commit

