

LAPORAN PRAKTIKUM
POSTTEST 1
ALGORITMA PEMROGRAMAN LANJUT

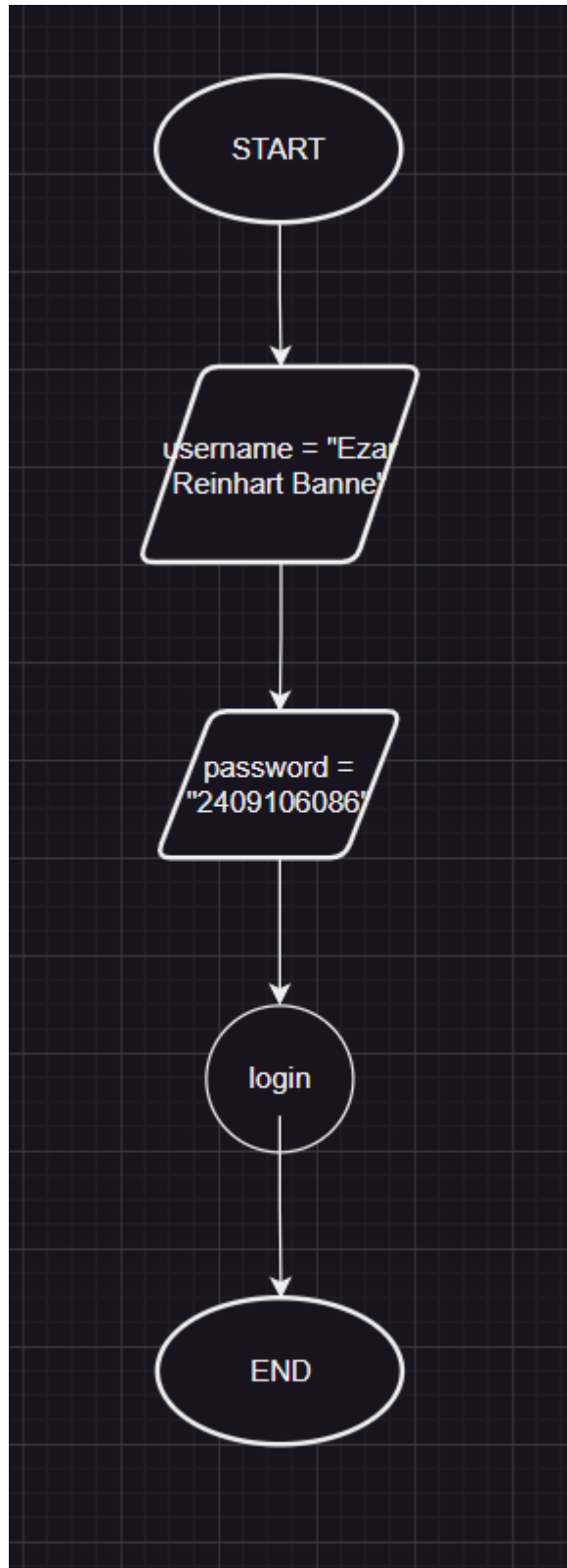


Disusun oleh:
Ahmad Habibi (2409106086)
Kelas (B2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

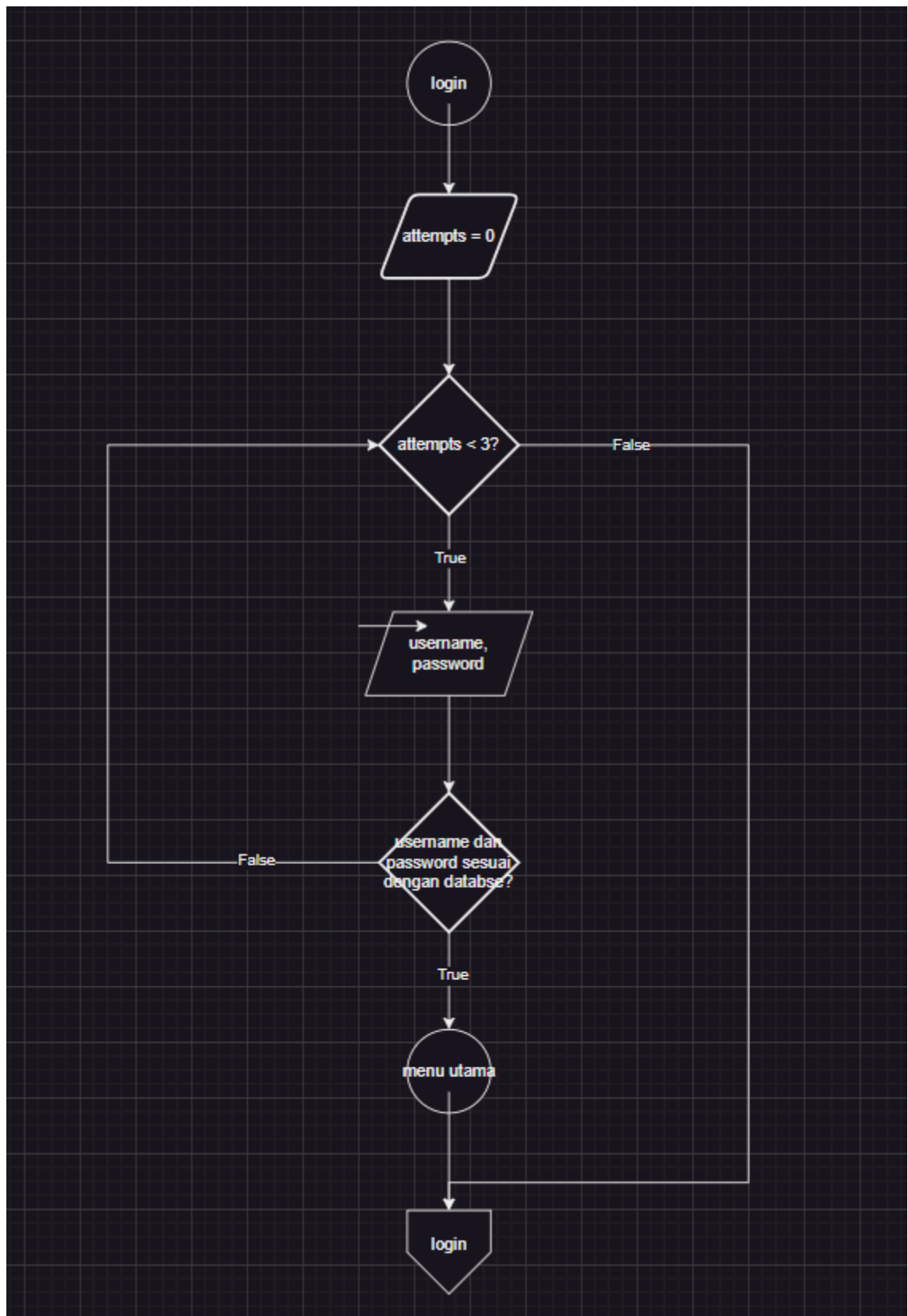
1. Flowchart

- Main



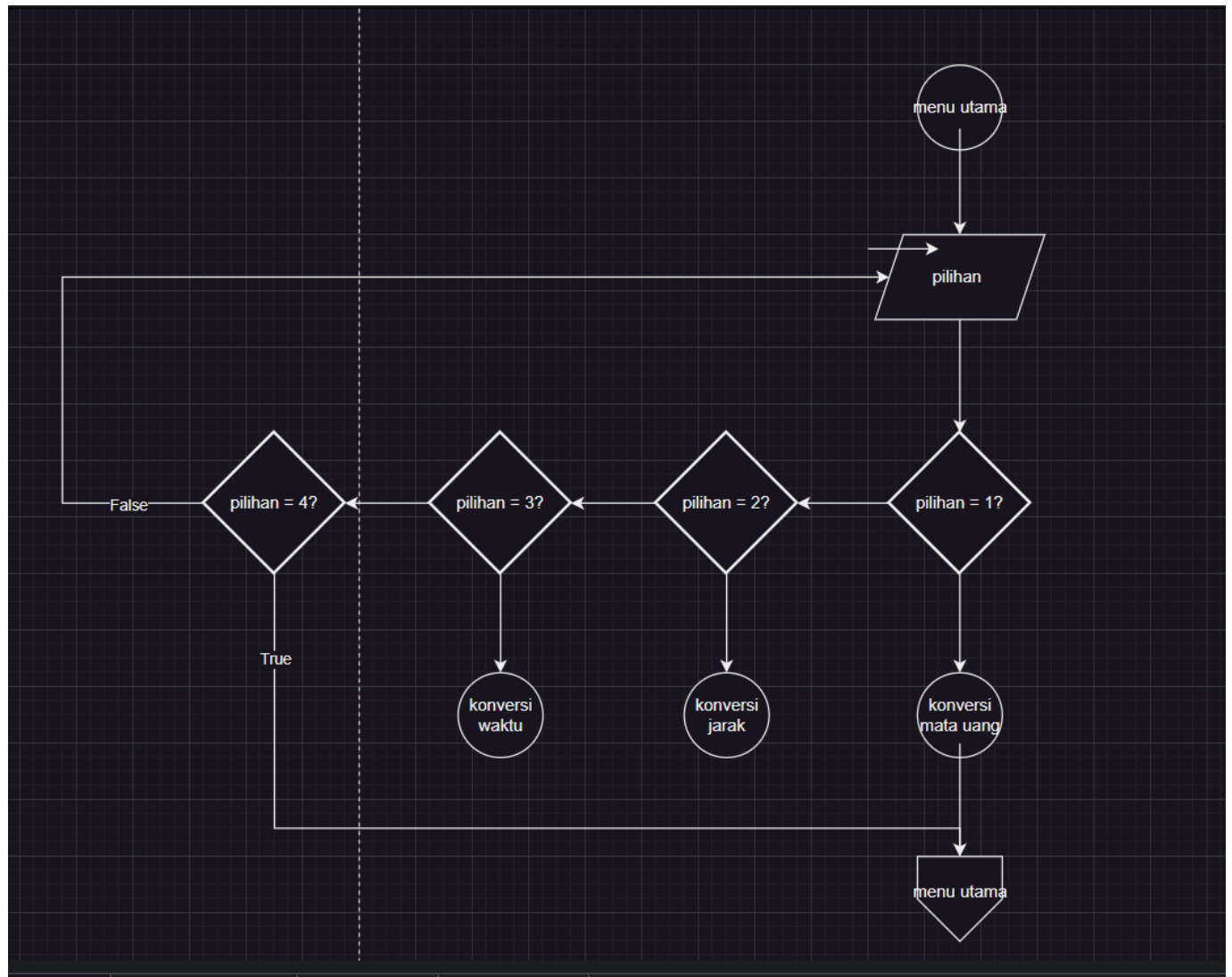
Gambar 1.1 Main

- Login



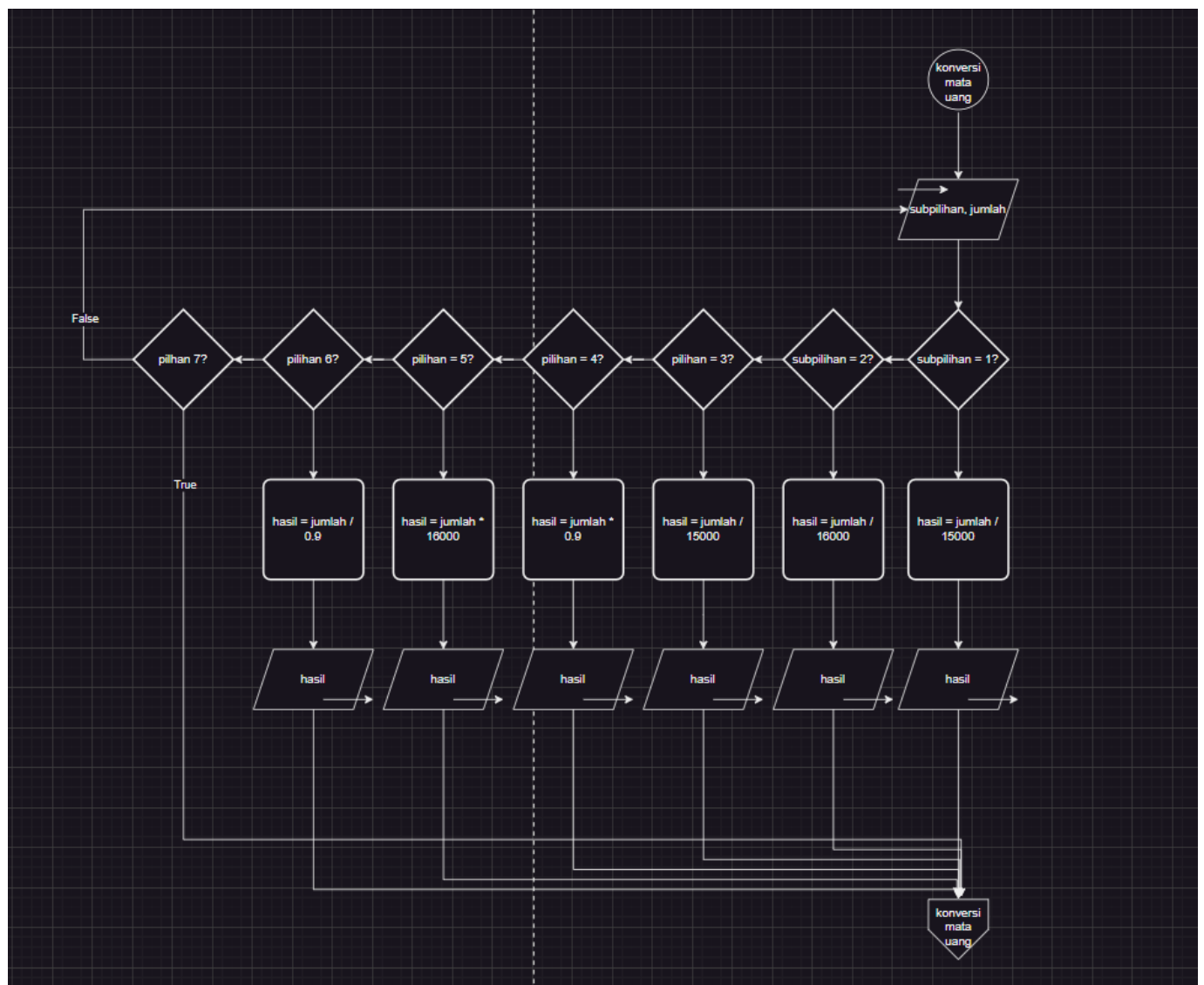
Gambar 1.2 Login

- Menu Utama



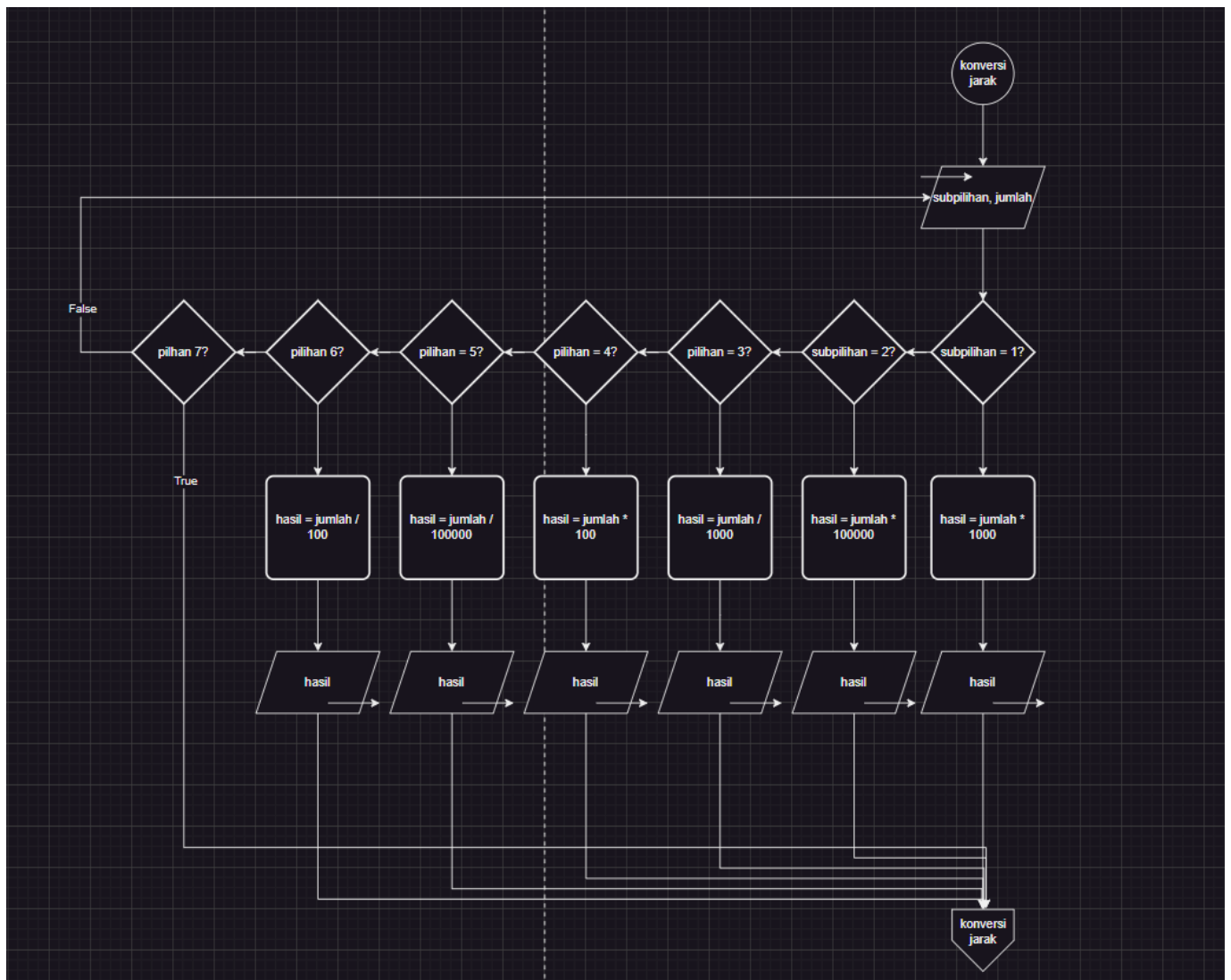
Gambar 1.3 Menu-Utama

- Konversi Mata uang



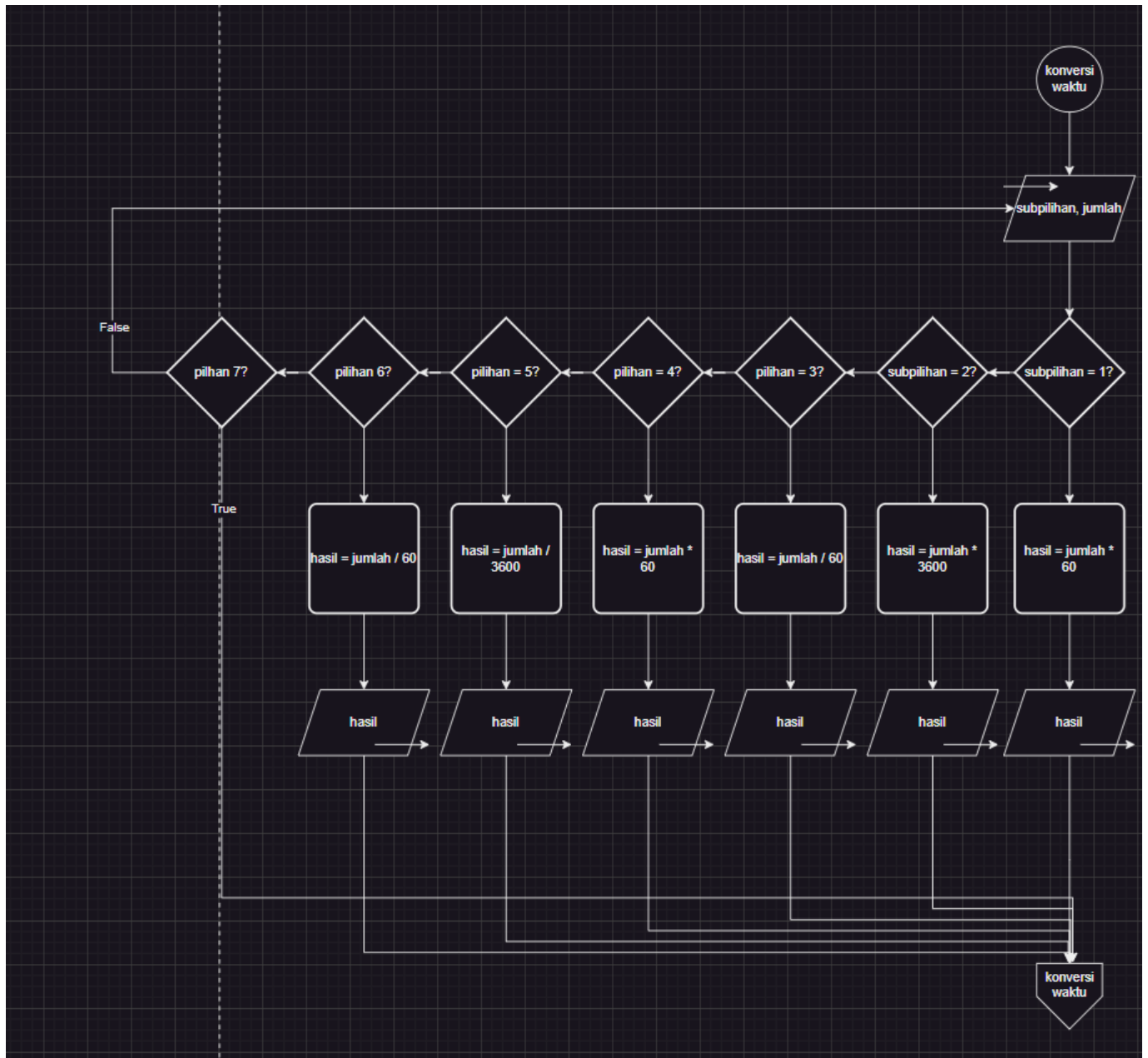
Gambar 1.4 Konversi-Mata-Uang

- Konversi Jarak



Gambar 1.5 Konversi-Jarak

- Konversi Waktu



Gambar 1.6 Konversi-Wakt

2. Analisis Program

2.1 Deskripsi Singkat Program

Program kali ini berfungsi untuk mengkonversi beberapa pilihan, yaitu:

1. Konversi mata uang
2. Konversi jarak
3. Konversi waktu

Program tidak akan berhenti hingga user meminta untuk berhenti.

2.2 Penjelasan Alur & Algoritma

Program dimulai dengan proses login, di mana pengguna diminta memasukkan username dan password. Jika kombinasi yang dimasukkan sesuai dengan kredensial yang telah ditentukan dalam fungsi `cek_login()`, maka login berhasil, dan pengguna bisa melanjutkan ke menu utama. Jika salah, pengguna diberi kesempatan hingga tiga kali sebelum program berhenti.

Setelah login berhasil, program menampilkan **menu utama** yang memiliki empat pilihan utama:

1. Konversi Mata Uang
2. Konversi Jarak
3. Konversi Waktu
4. Logout

Jika pengguna memilih salah satu dari tiga opsi pertama, mereka akan diberikan sub-menu dengan berbagai pilihan konversi. Program kemudian meminta pengguna untuk memasukkan jumlah yang akan dikonversi. Setelah itu, program memanggil fungsi yang sesuai (`konversi_mata_uang()`, `konversi_jarak()`, atau `konversi_waktu()`) untuk melakukan perhitungan dan menampilkan hasilnya.

Pengguna bisa terus melakukan konversi sampai mereka memilih opsi Logout, yang akan mengakhiri program dengan menampilkan pesan bahwa logout berhasil. Jika pengguna memasukkan pilihan yang tidak valid di menu utama atau sub-menu, program akan menampilkan pesan kesalahan dan meminta input ulang.

3. Source Code

```
1 bool cek_login(string username, string password) {
2     return (username == "ezar reinhart banne" && password == "2409106086");
3 }
4
5 bool login() {
6     string username, password;
7     int percobaan = 0;
8
9     while (percobaan < 3) {
10         cout << "Masukkan Username: ";
11         getline(cin, username);
12         cout << "Masukkan Password: ";
13         cin >> password;
14
15         if (cek_login(username, password)) {
16             cout << "Login Berhasil!\n\n";
17             return true;
18         }
19         percobaan++;
20         cout << "Login Gagal! Sisa percobaan: " << (3 - percobaan) << "\n\n";
21     }
22
23     cout << "Terlalu banyak percobaan gagal. Program berhenti." << endl;
24     return false;
25 }
26
27 double konversi_mata_uang(int pilihan, double jumlah) {
28     double hasil = 0;
29     if (pilihan == 1) hasil = jumlah * 0.000064; // Rupiah ke Dolar AS
30     else if (pilihan == 2) hasil = jumlah * 0.000059; // Rupiah ke Euro
31     else if (pilihan == 3) hasil = jumlah * 15625; // Dolar AS ke Rupiah
32     else if (pilihan == 4) hasil = jumlah * 0.92; // Dolar AS ke Euro
33     else if (pilihan == 5) hasil = jumlah * 16800; // Euro ke Rupiah
34     else if (pilihan == 6) hasil = jumlah * 1.09; // Euro ke Dolar AS
35     else cout << "Pilihan tidak valid!\n";
36     return hasil;
37 }
38
39 double konversi_jarak(int pilihan, double jumlah) {
40     double hasil = 0;
41     if (pilihan == 1) hasil = jumlah * 1000; // Kilometer ke Meter
42     else if (pilihan == 2) hasil = jumlah * 100000; // Kilometer ke Centimeter
43     else if (pilihan == 3) hasil = jumlah / 1000; // Meter ke Kilometer
44     else if (pilihan == 4) hasil = jumlah * 100; // Meter ke Centimeter
45     else if (pilihan == 5) hasil = jumlah / 100000; // Centimeter ke Kilometer
46     else if (pilihan == 6) hasil = jumlah / 100; // Centimeter ke Meter
47     else cout << "Pilihan tidak valid!\n";
48     return hasil;
49 }
50
51 double konversi_waktu(int pilihan, double jumlah) {
52     double hasil = 0;
53     if (pilihan == 1) hasil = jumlah * 60; // Jam ke Menit
54     else if (pilihan == 2) hasil = jumlah * 3600; // Jam ke Detik
55     else if (pilihan == 3) hasil = jumlah / 60; // Menit ke Jam
56     else if (pilihan == 4) hasil = jumlah * 60; // Menit ke Detik
57     else if (pilihan == 5) hasil = jumlah / 3600; // Detik ke Jam
58     else if (pilihan == 6) hasil = jumlah / 60; // Detik ke Menit
59     else cout << "Pilihan tidak valid!\n";
60     return hasil;
61 }
62
63 void menu_utama() {
64     int pilihan;
65     do {
66         cout << "\nMenu Utama:\n1. Konversi Mata Uang\n2. Konversi Jarak\n3. Konversi Waktu\n4. Logout\nPilih menu: ";
67         cin >> pilihan;
68
69         if (pilihan >= 1 && pilihan <= 3) {
70             double jumlah;
71             int sub_pilihan;
72
73             if (pilihan == 1) {
74                 cout << "\nKonversi Mata Uang:\n1. Rupiah ke Dolar AS\n2. Rupiah ke Euro\n3. Dolar AS ke Rupiah\n4. Dolar AS ke Euro\n5. Euro ke Rupiah\n6. Euro ke Dolar AS\nPilih opsi: ";
75             } else if (pilihan == 2) {
76                 cout << "\nKonversi Jarak:\n1. Kilometer ke Meter\n2. Kilometer ke Centimeter\n3. Meter ke Kilometer\n4. Meter ke Centimeter\n5. Centimeter ke Kilometer\n6. Centimeter ke Meter\nPilih opsi: ";
77             } else {
78                 cout << "\nKonversi Waktu:\n1. Jam ke Menit\n2. Jam ke Detik\n3. Menit ke Jam\n4. Menit ke Detik\n5. Detik ke Jam\n6. Detik ke Menit\nPilih opsi: ";
79             }
80
81             cin >> sub_pilihan;
82             cout << "Masukkan jumlah: ";
83             cin >> jumlah;
84
85             double hasil = (pilihan == 1) ? konversi_mata_uang(sub_pilihan, jumlah) :
86                 (pilihan == 2) ? konversi_jarak(sub_pilihan, jumlah) :
87                 konversi_waktu(sub_pilihan, jumlah);
88
89             if (hasil != -1) {
90                 cout << "Hasil konversi: " << hasil << "\n";
91             }
92             } else if (pilihan != 4) {
93                 cout << "Pilihan tidak valid!\n";
94             }
95         } while (pilihan != 4);
96
97         cout << "Logout berhasil. Program berhenti.\n";
98     }
99
100 int main() {
101     if (login()) menu_utama();
102     return 0;
103 }
```

Gambar 3.1

4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. Skenario 1

Pada skenario ini program dijalankan senormalnya untuk mengkonversi mata uang Rupiah ke Euro

```
Masukkan Username: ezar reinhart banne  
Masukkan Password: 2409106086  
Login Berhasil!
```

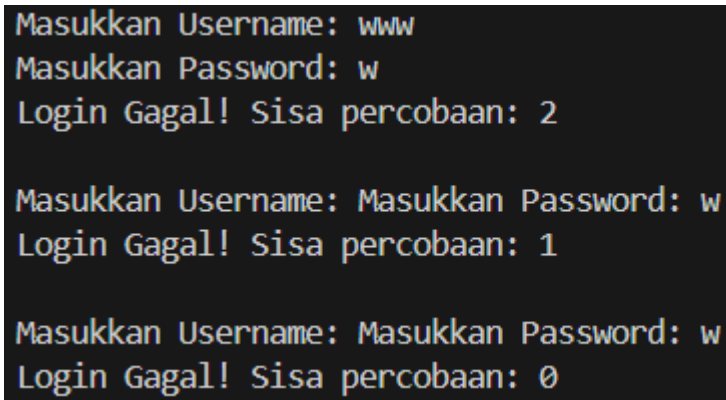
```
Menu Utama:  
1. Konversi Mata Uang  
2. Konversi Jarak  
3. Konversi Waktu  
4. Logout  
Pilih menu: 1
```

```
Konversi Mata Uang:  
1. Rupiah ke Dolar AS  
2. Rupiah ke Euro  
3. Dolar AS ke Rupiah  
4. Dolar AS ke Euro  
5. Euro ke Rupiah  
6. Euro ke Dolar AS  
Pilih opsi: 2  
Masukkan jumlah: 3000000  
Hasil konversi: 177
```

Gambar 4,1 Hasil-Uji-Coba-Berhasil

2. Skenario 2

Pada skenario ini, user menginput username ataupun password yang salah sehingga program otomatis berhenti.



```
Masukkan Username: www
Masukkan Password: w
Login Gagal! Sisa percobaan: 2

Masukkan Username: Masukkan Password: w
Login Gagal! Sisa percobaan: 1

Masukkan Username: Masukkan Password: w
Login Gagal! Sisa percobaan: 0
```

Gambar 4.2 Hasil-Uji-Coba-Gagal

4.2 Hasil Output

Contoh hasil output dari perhitungan Konversi Mata uang Rupiah ke Euro.

```
Masukkan Username: ezar reinhart banne  
Masukkan Password: 2409106086  
Login Berhasil!
```

```
Menu Utama:  
1. Konversi Mata Uang  
2. Konversi Jarak  
3. Konversi Waktu  
4. Logout  
Pilih menu: 1
```

```
Konversi Mata Uang:  
1. Rupiah ke Dolar AS  
2. Rupiah ke Euro  
3. Dolar AS ke Rupiah  
4. Dolar AS ke Euro  
5. Euro ke Rupiah  
6. Euro ke Dolar AS  
Pilih opsi: 2  
Masukkan jumlah: 3000000  
Hasil konversi: 177
```

```
Menu Utama:  
1. Konversi Mata Uang  
2. Konversi Jarak  
3. Konversi Waktu  
4. Logout  
Pilih menu: 4  
Logout berhasil. Program berhenti.
```

Gambar 4.3 Output

5. Git

1. Inisialisasi Repo git

```
PS D:\dump tugas\project 0\praktikum-apl> git init
Reinitialized existing Git repository in D:/dump tugas/project 0/praktikum-apl/.git/
```

Gambar 5.1 Inisialisasi-Repository

Pada gambar 5.1, perintah git init dijalankan di dalam direktori D:\dump tugas\project 0\praktikum-apl, yang menghasilkan pesan bahwa Git telah berhasil diinisialisasi dalam folder tersebut. Jika pertama kali menjalankan perintah ini, Git akan membuat folder tersembunyi bernama .git, yang berisi semua informasi terkait version control dari proyek tersebut.

2. Menambahkan File untuk dicommit

```
PS D:\dump tugas\project 0\praktikum-apl> git add .
```

Gambar 5.2 Menambahkan-File

Setelah repository Git diinisialisasi, langkah berikutnya adalah menambahkan file ke dalam staging area. Ini adalah tahap sebelum file benar-benar disimpan dalam repository Git. Perintah yang digunakan untuk menambahkan file ke staging area. Pada gambar 5.2, perintah git add . digunakan. Titik (.) dalam perintah ini berarti semua file dalam direktori saat ini akan ditambahkan ke staging area.

3. Commit

```
PS D:\dump tugas\project 0\praktikum-apl> git commit -m "Finish Post Test 1"
[main 38ac0bc] Finish Post Test 1
3 files changed, 0 insertions(+), 0 deletions(-)
rename post-test/post-test-apl-1/{24091060104_AhmadHabibi-PT-1.cpp => 24091060104-AhmadHabibi-PT-1.cpp} (100%)
rename post-test/post-test-apl-1/{24091060104_AhmadHabibi-PT-1.exe => 24091060104-AhmadHabibi-PT-1.exe} (100%)
rename post-test/post-test-apl-1/{24091060104_AhmadHabibi-PT-1.cpp => 24091060104-AhmadHabibi-PT-1.cpp} (100%)
rename post-test/post-test-apl-1/{24091060104_AhmadHabibi-PT-1.exe => 24091060104-AhmadHabibi-PT-1.exe} (100%)
rename post-test/post-test-apl-1/{24091060104_AhmadHabibi-PT-1.pdf => 24091060104-AhmadHabibi-PT-1.pdf} (100%)
```

Gambar 5.3 Commit

Commit adalah proses menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository Git. Commit membantu mendokumentasikan perubahan yang dilakukan pada proyek dengan memberikan pesan deskriptif. Pada gambar 5.3, perintah git commit -m "Finish Post Test 1" digunakan untuk menyimpan perubahan dengan pesan commit "Finish Post Test 1".

4. push untuk mengupload file ke github

```
PS D:\dump tugas\project 0\praktikum-apl> git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 20 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 353 bytes | 176.00 KiB/s, done.
Total 4 (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/aeBeeBoy/praktikum-apl.git
    c5bfa10..38ac0bc  main -> main
PS D:\dump tugas\project 0\praktikum-apl>
```

Gambar 5.4 Push

Setelah melakukan commit, langkah selanjutnya adalah mengunggah perubahan dari repository lokal ke repository GitHub. Pada gambar 5.4, perintah git push digunakan untuk mengunggah commit terbaru ke repository GitHub. Perintah ini akan mengunggah perubahan ke *branch* yang aktif (default-nya adalah main atau master).