

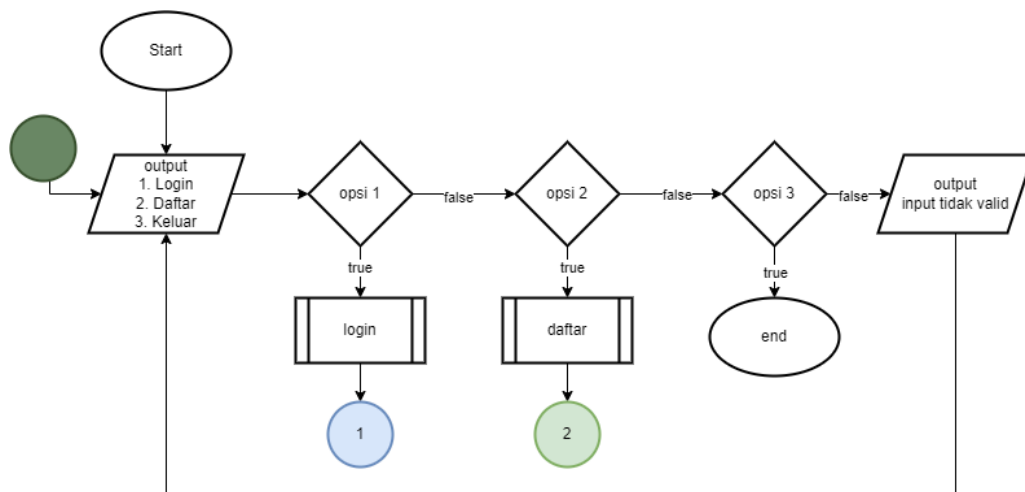
LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN LANJUT

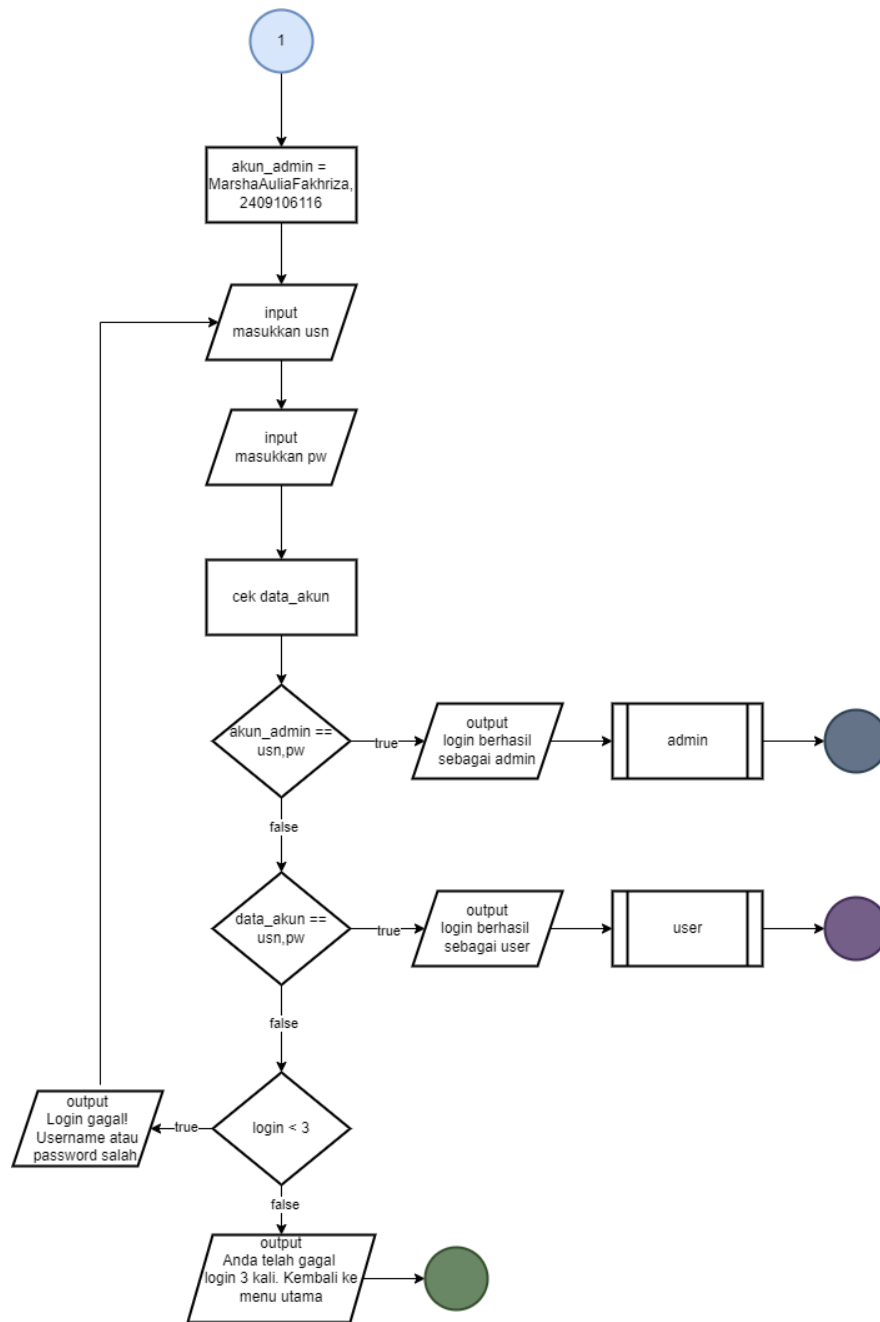


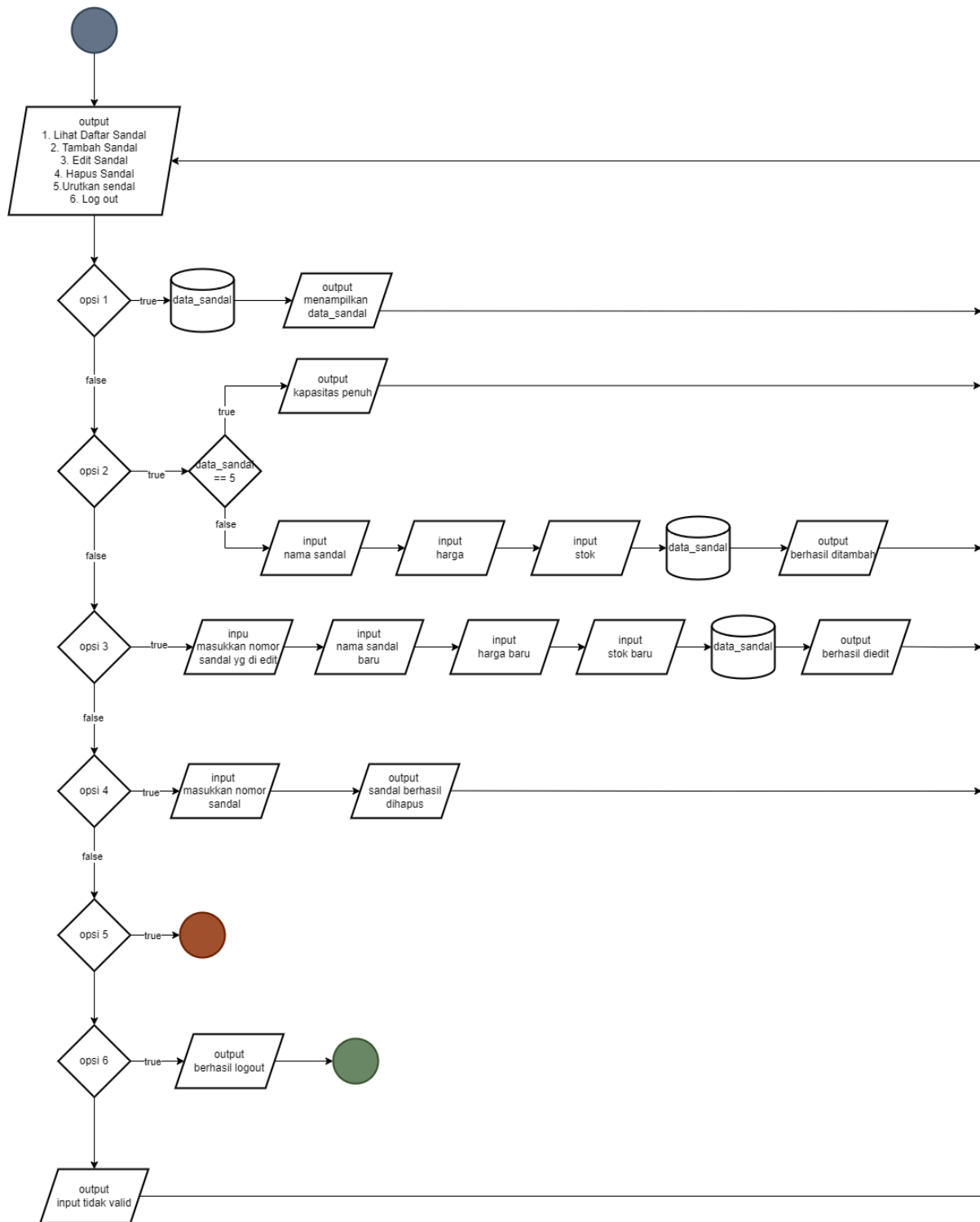
Disusun oleh:
Marsha Aulia Fakhriza (2409106116)
Kelas (C'24)

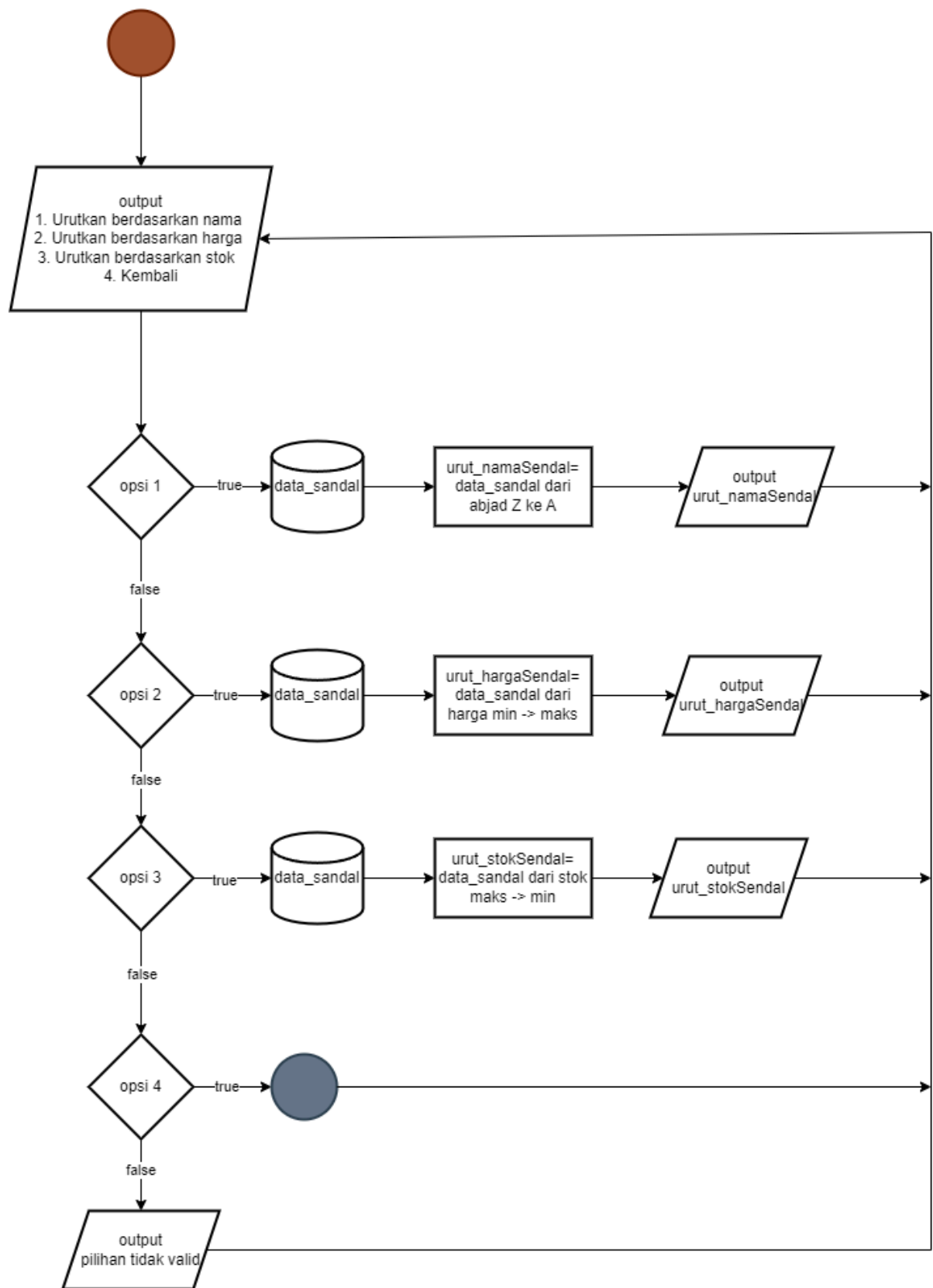
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

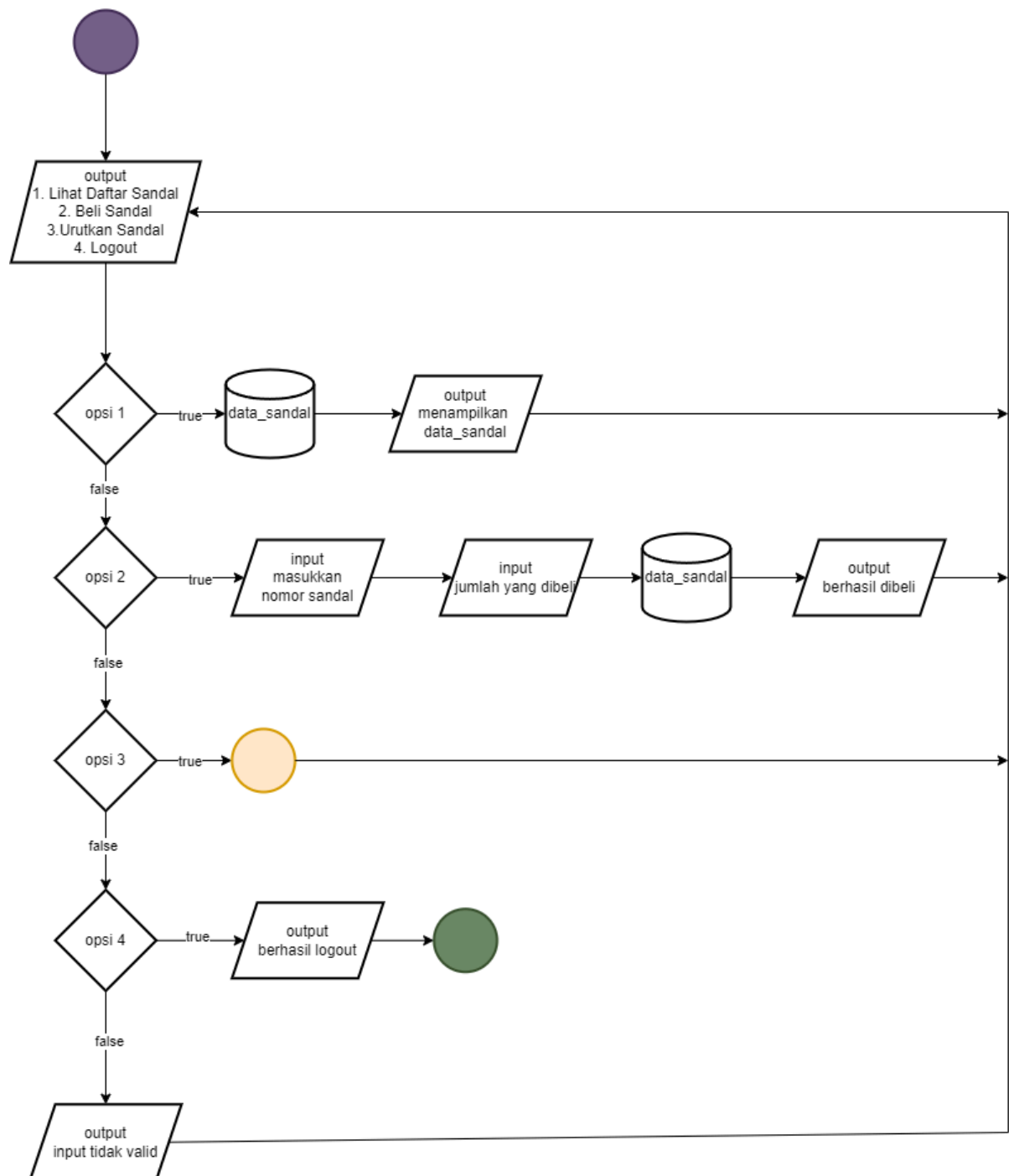
1. Flowchart

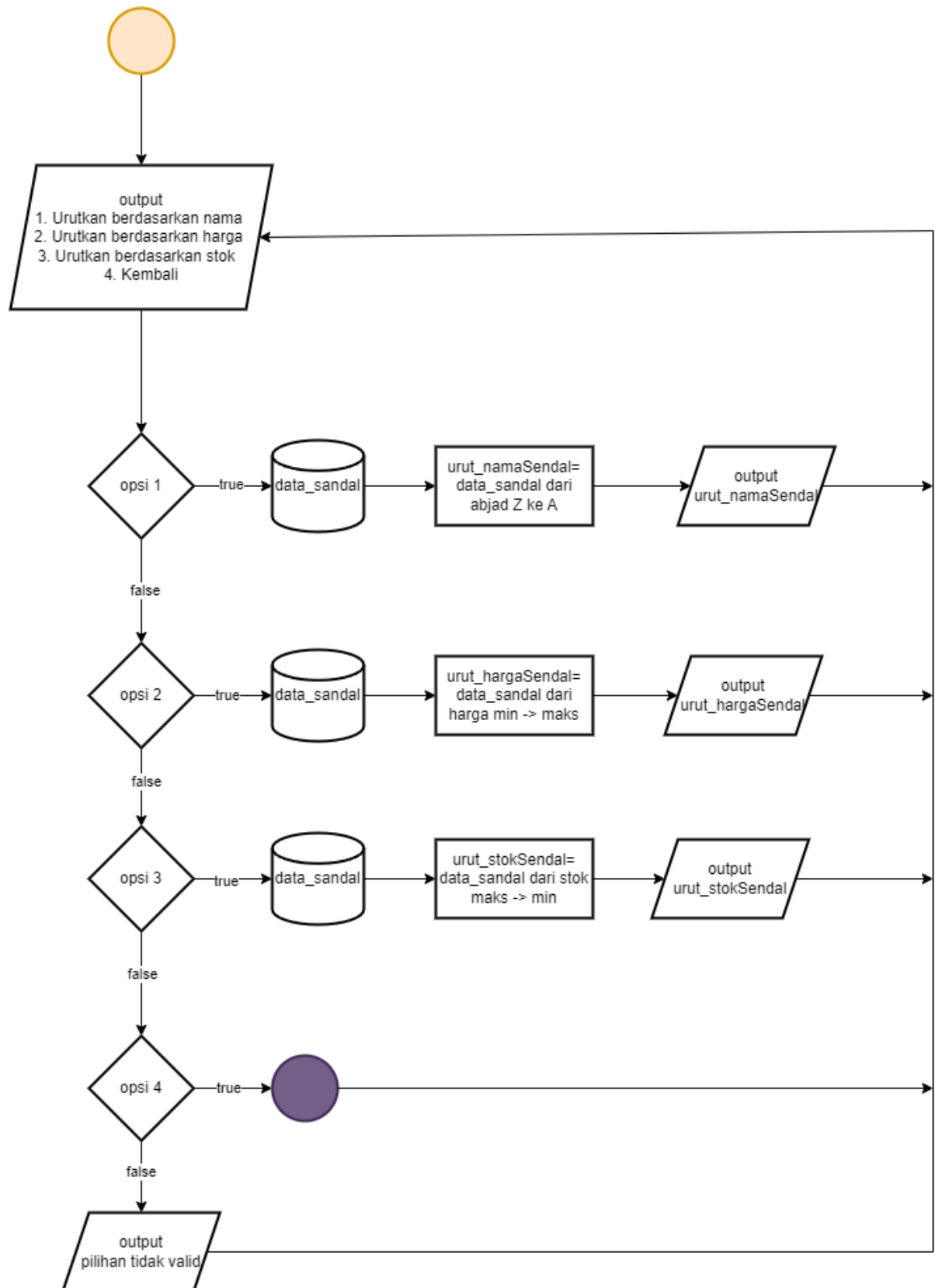


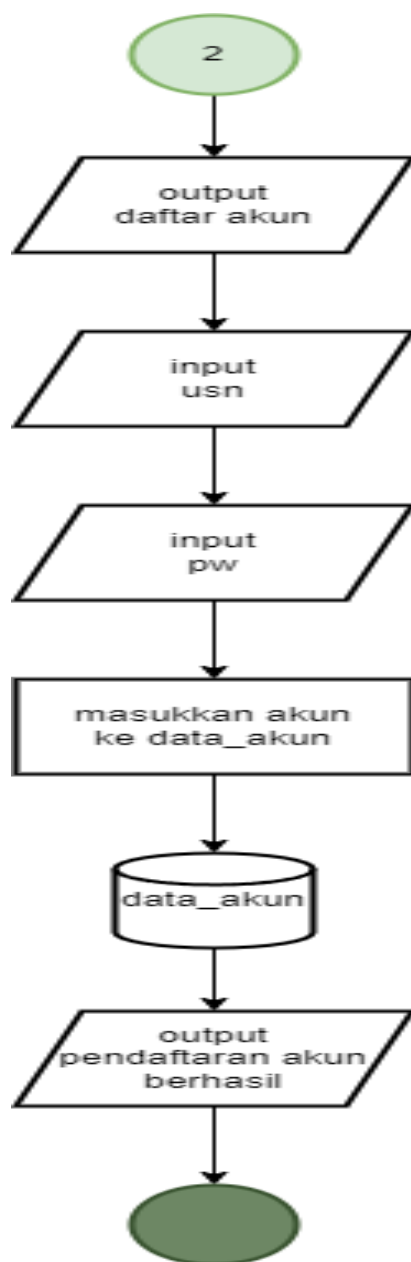












2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini merupakan sistem manajemen pembelian sandal yang memvalidasi dan mengelola sandal setelah pengguna berhasil login. Sistem ini dirancang untuk dua jenis pengguna dengan hak akses berbeda: Admin dan Pembeli.

Pada posttest ini tidak merubah seluruh source code hanya saja terdapat tambahan fitur yaitu mengurutkan nama berdasarkan abjad dari Z ke A (descending) dimana fitur ini menggunakan algoritma Bubble sort, lalu terdapat mengurutkan harga berdasarkan dari harga terendah ke tertinggi (ascending) menggunakan algoritma Insertion sort dan fitur yang terakhir yaitu mengurutkan stok berdasarkan dari stok tertinggi ke terendah menggunakan algoritma Selection sort.

Sesuai tugas dari posttest karena nilai absen saya genap maka untuk descending harus huruf ,untuk ascending berupa angka dan menggunakan tiga konsep algoritma.

2.2 Penjelasan Alur & Algoritma

1 Login & Pendaftaran Pengguna

- Saat program dijalankan, pengguna disajikan dengan menu utama yang berisi tiga pilihan: Login, Daftar, dan Keluar.
- Jika pengguna memilih Login, mereka harus memasukkan username dan password.
- Program akan mengecek kredensial pengguna dalam daftar pengguna yang tersimpan. Jika cocok, pengguna akan masuk sesuai dengan role mereka (admin atau pembeli).
- Jika pengguna salah memasukkan password sebanyak 3 kali, mereka akan dikembalikan ke menu utama.
- Jika pengguna memilih Daftar, mereka dapat membuat akun baru dengan peran otomatis sebagai pembeli (admin tidak bisa didaftarkan langsung melalui program).

2 Fitur Admin

- **Melihat Daftar Sandal**

Admin dapat melihat daftar semua sandal yang tersedia, beserta informasi seperti:

1. Nama sandal
2. Harga sandal
3. Jumlah stok sandal
4. Hapus sandal
5. Urutkan sandal
6. Logout

- **Menambah Sandal Baru**

Admin bisa menambahkan sandal baru dengan memasukkan nama, harga, dan stok. Namun, ada batasan maksimal 5 sandal dalam daftar. Jika kapasitas penuh, admin tidak bisa menambahkan lagi.

- **Mengedit Data Sandal**

Admin dapat mengubah data sandal yang sudah ada dengan memasukkan nomor sandal yang ingin diedit, lalu mengganti nama, harga, atau stoknya.

- **Menghapus Sandal dari Daftar**

Admin dapat menghapus sandal berdasarkan nomor yang dipilih. Sebelum menghapus, program akan meminta konfirmasi dari admin.

- **Urutkan Sandal**

Pada fitur ini terdapat pilihan untuk mengurutkan sandal yang pertama mengurutkan berdasarkan nama dari abjad Z ke A, lalu yang kedua berdasarkan harga terendah ke tertinggi dan yang terakhir berdasarkan stok dari tertinggi ke terendah

3 Fitur Pembeli

Jika pengguna yang login memiliki role sebagai pembeli, mereka memiliki akses ke fitur-fitur berikut:

- a. Melihat Daftar Sandal Pembeli dapat melihat daftar sandal yang tersedia, beserta informasi harga dan stok.
- b. Membeli Sandal Pembeli dapat membeli sandal dengan memasukkan nomor sandal dan jumlah yang ingin dibeli. Program akan melakukan pengecekan:

- Jika stok cukup, transaksi berhasil dan stok akan berkurang.
 - Jika stok tidak mencukupi, program akan menampilkan pesan kesalahan.
 - Jika barang yang dibeli lebih dari 3 maka mendapatkan diskon
- c. Fitur Urutkan sandal sama seperti admin terdapat tiga pilihan untuk melihatnya berdasarkan nama lalu harga dan stok.

3. Source Code

Saya hanya mengambil source code yang baru saja dibuat dengan mengikuti peraturan posttest 6

A. Deklarasi Function Fitur Baru Menggunakan Algoritma

Pada potongan kode ini function harus dideklarasikan dahulu sebelum membuat function nya

Source Code:

```
void sortSandalByNameDesc(Sandal daftarSandal[], int jumlahSandal);
void sortSandalByPriceAsc(Sandal daftarSandal[], int jumlahSandal);
void sortSandalByStockDesc(Sandal daftarSandal[], int jumlahSandal);
void menuUrutkan(Sandal daftarSandal[], int jumlahSandal);
```

B. Function Sorting Nama Descending

Pada function ini data sandal akan di sorting dari abjad Z ke A menggunakan algoritma bubble sort

Source code

```
void sortSandalByNameDesc(Sandal daftarSandal[], int jumlahSandal) {
    for (int i = 0; i < jumlahSandal - 1; i++) {
        for (int j = 0; j < jumlahSandal - i - 1; j++) {
            if (daftarSandal[j].nama < daftarSandal[j + 1].nama) {
                Sandal temp = daftarSandal[j];
                daftarSandal[j] = daftarSandal[j + 1];
                daftarSandal[j + 1] = temp;
            }
        }
    }
}
```

```
}
```

C. Function Sorting Harga Ascending

Potongan kode ini Sorting harga ascending yaitu harga terendah ke tertinggi menggunakan insertion sort

Source code

```
void sortSandalByPriceAsc(Sandal daftarSandal[], int jumlahSandal) {  
    for (int i = 1; i < jumlahSandal; i++) {  
        Sandal key = daftarSandal[i];  
        int j = i - 1;  
  
        while (j >= 0 && daftarSandal[j].harga > key.harga) {  
            daftarSandal[j + 1] = daftarSandal[j];  
            j--;  
        }  
  
        daftarSandal[j + 1] = key;  
    }  
}
```

D. Function Sorting Stok Descending

Potongan kode ini Sorting stok descending yaitu stok tertinggi ke terendah menggunakan selection sort

Source code

```
void sortSandalByStockDesc(Sandal daftarSandal[], int jumlahSandal) {  
    for (int i = 0; i < jumlahSandal - 1; i++) {  
        int max_idx = i;  
        for (int j = i + 1; j < jumlahSandal; j++) {
```

```

        if (daftarSandal[j].stok > daftarSandal[max_idx].stok) {
            max_idx = j;
        }
    }
    if (max_idx != i) {
        Sandal temp = daftarSandal[i];
        daftarSandal[i] = daftarSandal[max_idx];
        daftarSandal[max_idx] = temp;
    }
}
}
}

```

E. Function Menu Fitur Baru

Kode ini untuk memilih mau mengurutkan berdasarkan apa sesuai kemauan kita

Source code

```

void menuUrutkan(Sandal daftarSandal[], int jumlahSandal) {
    int pilihanUrut;
    cout << "\n----- MENU PENGURUTAN SANDAL -----" << endl;
    cout << "1. Urutkan berdasarkan nama (Z-A)" << endl;
    cout << "2. Urutkan berdasarkan harga (Rendah-Tinggi)" << endl;
    cout << "3. Urutkan berdasarkan stok (Tinggi-Rendah)" << endl;
    cout << "4. Kembali" << endl;
    cout << "Pilihan: ";
    cin >> pilihanUrut;

    switch (pilihanUrut) {
        case 1:
            sortSandalByNameDesc(daftarSandal, jumlahSandal);
            cout << "Sandal telah diurutkan berdasarkan nama secara menurun (Z-A)!" << endl;
            tampilkanDaftarSandal(daftarSandal, jumlahSandal);
            break;
        case 2:
            sortSandalByPriceAsc(daftarSandal, jumlahSandal);
            cout << "Sandal telah diurutkan berdasarkan harga dari terendah ke tertinggi!" << endl;
            tampilkanDaftarSandal(daftarSandal, jumlahSandal);
            break;
        case 3:
            sortSandalByStockDesc(daftarSandal, jumlahSandal);
            cout << "Sandal telah diurutkan berdasarkan stok dari

```

```
    terbanyak!" << endl;
    tampilkanDaftarSandal(daftarSandal, jumlahSandal);
    break;
case 4:
    cout << "Kembali ke menu utama." << endl;
    break;
default:
    cout << "Pilihan tidak valid!" << endl;
    break;
}
```

4. Uji Coba dan Hasil Output

Hasil Uji coba yang saya ambil hanya function yang baru Penerapan Algoritma

4.1 Uji Coba

1. Pengujian Mengurutkan Berdasarkan Nama

input : Menginput pilihan 1 untuk mengurutkan berdasarkan nama

output yang diharapkan : Berhasil menampilkan data sandal yang diurutkan

2. Pengujian Mengurutkan Berdasarkan Harga

input : Menginput pilihan 2 untuk mengurutkan berdasarkan Harga

output yang diharapkan : Berhasil menampilkan data sandal yang diurutkan

3. Pengujian Mengurutkan Berdasarkan Stok

input : Menginput pilihan 3 untuk mengurutkan berdasarkan Stok

output yang diharapkan : Berhasil menampilkan data sandal yang diurutkan

4.2 Hasil Output

```
----- MENU PENGURUTAN SANDAL -----
1. Urutkan berdasarkan nama (Z-A)
2. Urutkan berdasarkan harga (Rendah-Tinggi)
3. Urutkan berdasarkan stok (Tinggi-Rendah)
4. Kembali
Pilihan: 1
Sandal telah diurutkan berdasarkan nama secara menurun (Z-A)!

Daftar Sandal:
-----
No | Nama          | Harga | Stok
-----
1  | Sandal Sport  | 60000 | 7
2  | Sandal Rumah  | 30000 | 15
3  | Sandal Kulit  | 75000 | 8
4  | Sandal Jepit  | 20000 | 10
5  | Sandal Gunung | 50000 | 5

----- MENU PEMBELI -----
1. Lihat Daftar Sandal
2. Beli Sandal
3. Urutkan Sandal
4. Logout
Pilihan: █
```

Gambar 4.1.1

```
----- MENU PEMBELI -----
1. Lihat Daftar Sandal
2. Beli Sandal
3. Urutkan Sandal
4. Logout
Pilihan: 3

----- MENU PENGURUTAN SANDAL -----
1. Urutkan berdasarkan nama (Z-A)
2. Urutkan berdasarkan harga (Rendah-Tinggi)
3. Urutkan berdasarkan stok (Tinggi-Rendah)
4. Kembali
Pilihan: 2
Sandal telah diurutkan berdasarkan harga dari terendah ke tertinggi!

Daftar Sandal:
-----
No | Nama          | Harga | Stok
-----
1  | Sandal Jepit  | 20000 | 10
2  | Sandal Rumah  | 30000 | 15
3  | Sandal Gunung | 50000 | 5
4  | Sandal Sport  | 60000 | 7
5  | Sandal Kulit  | 75000 | 8
```

Gambar 4.1.2

```

----- MENU PEMBELI -----
1. Lihat Daftar Sandal
2. Beli Sandal
3. Urutkan Sandal
4. Logout
Pilihan: 3

----- MENU PENGURUTAN SANDAL -----
1. Urutkan berdasarkan nama (Z-A)
2. Urutkan berdasarkan harga (Rendah-Tinggi)
3. Urutkan berdasarkan stok (Tinggi-Rendah)
4. Kembali
Pilihan: 3
Sandal telah diurutkan berdasarkan stok dari terbanyak!

Daftar Sandal:
-----
No | Nama          | Harga | Stok
-----
1  | Sandal Rumah  | 30000 | 15
2  | Sandal Jepit  | 20000 | 10
3  | Sandal Kulit  | 75000 | 8
4  | Sandal Sport  | 60000 | 7
5  | Sandal Gunung | 50000 | 5

```

Gambar 4.1.3