

LAPORAN PRAKTIKUM ALGORITMA PEMOGRAMAN LANJUT



NAMA:Muhammad febriansyah

KELAS:B2 24 informatika

NIM:2409106090

TUGAS:laporan posttes 5

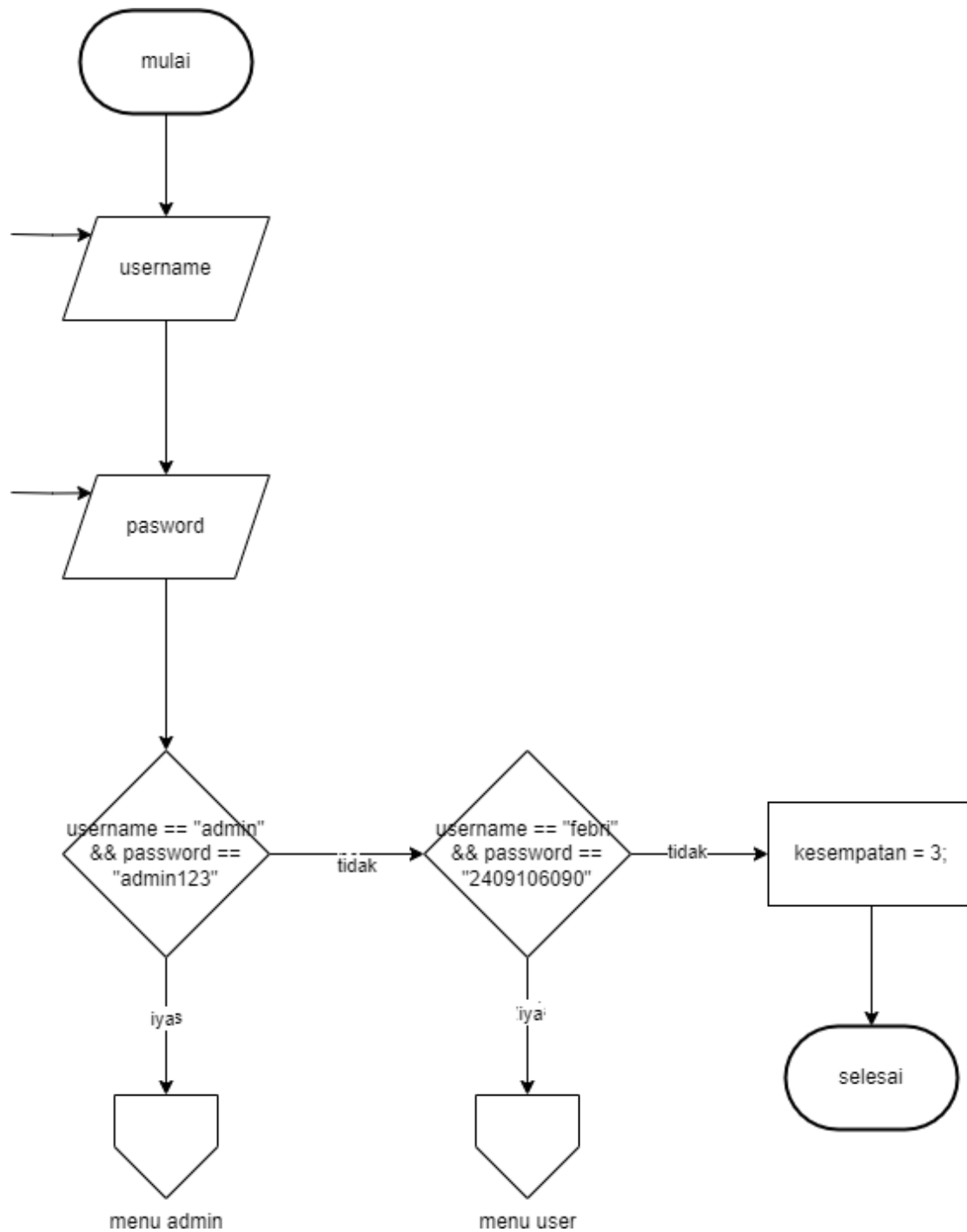
PROGRAM STUDI INFORMATIKA

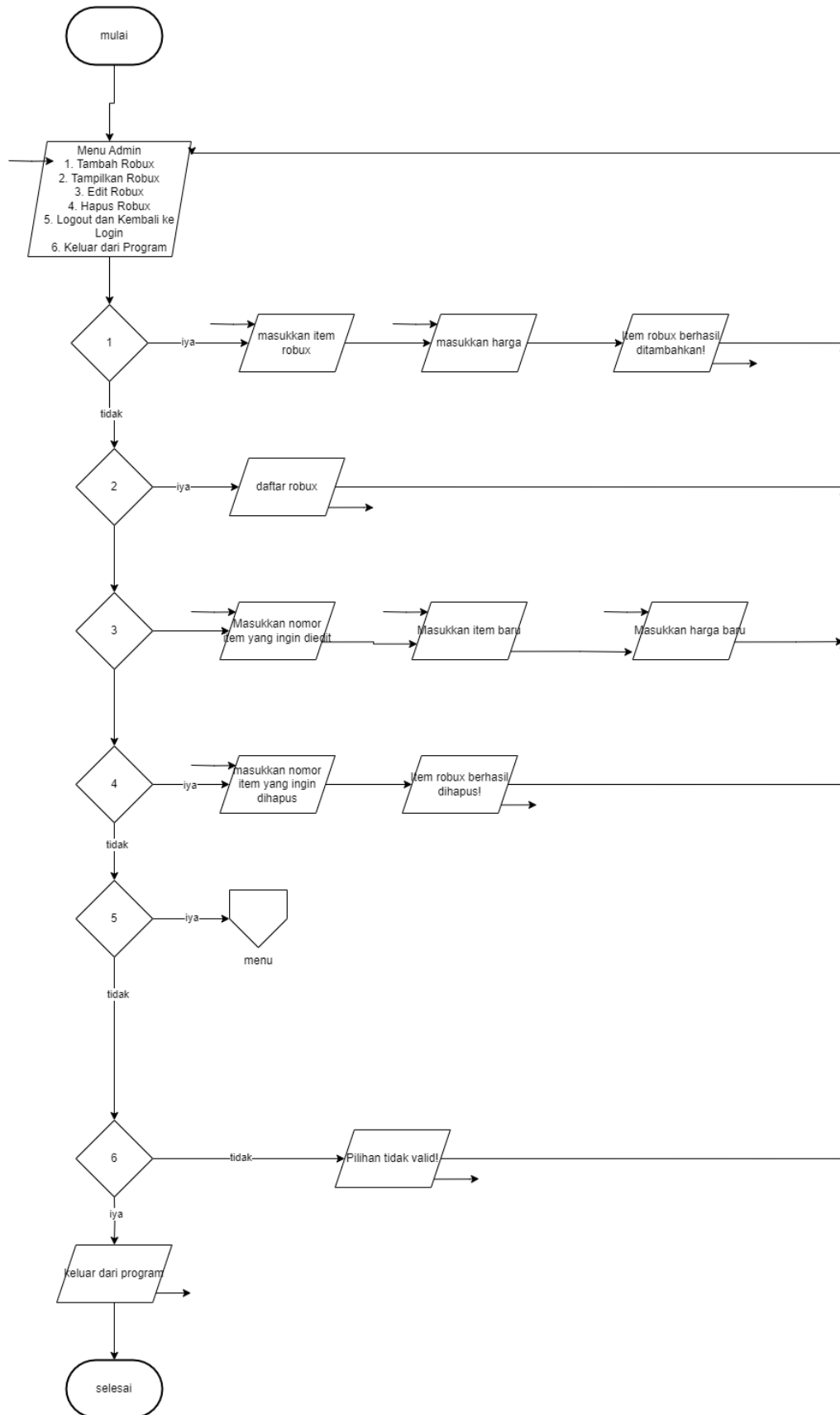
UNIVERSITAS MULAWARMAN

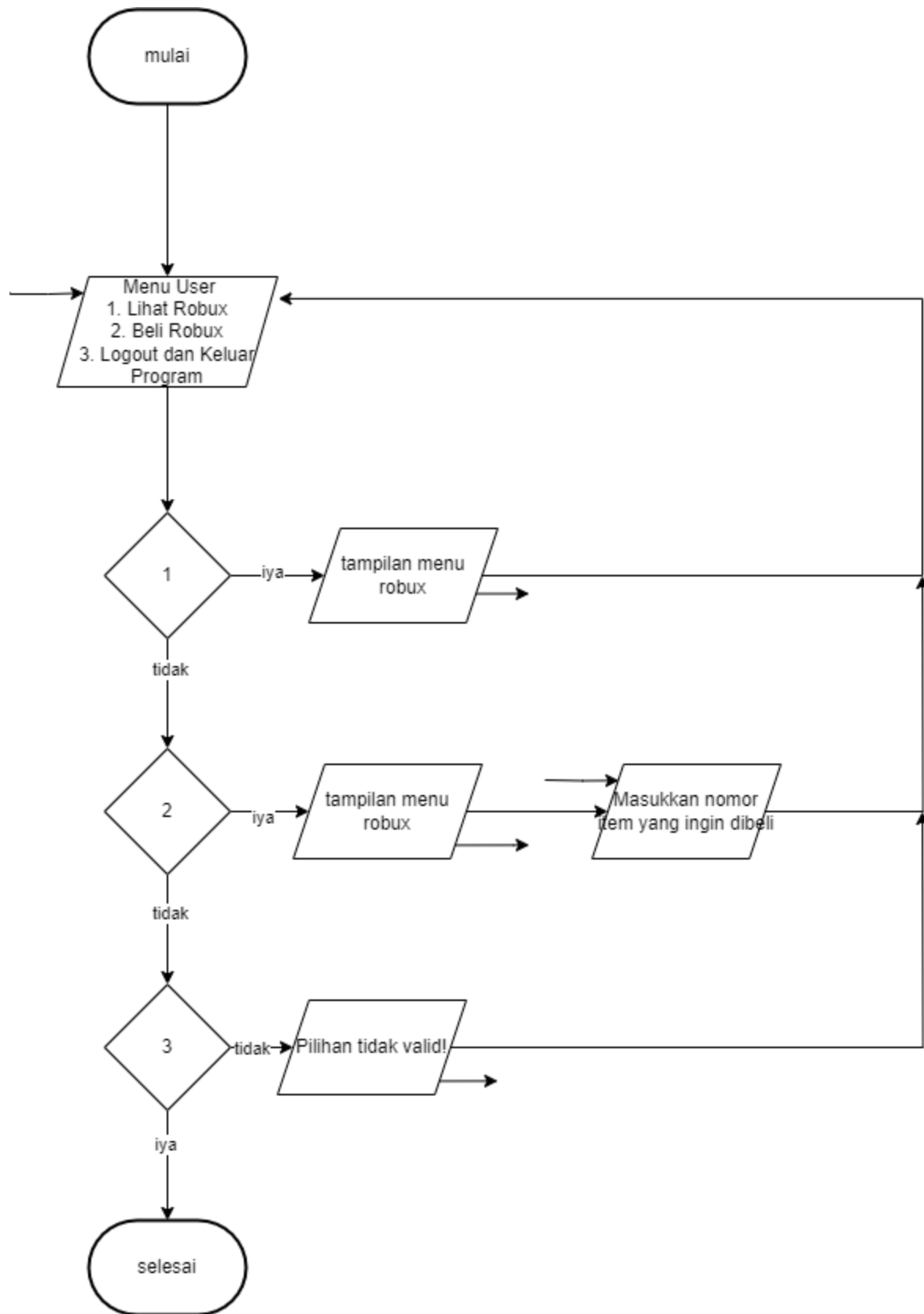
SAMARINDA

2025

1.FLOWCHART:







2. ANALISIS PROGRAM

Program ini adalah sistem manajemen penjualan Robux berbasis CLI (Command Line Interface). Program memiliki dua jenis pengguna:

Admin: Dapat menambahkan, menampilkan, mengedit, dan menghapus item Robux.

User: Hanya dapat melihat daftar item Robux dan melakukan pembelian

Sistem Login: Memastikan hanya pengguna yang memiliki kredensial yang benar dapat mengakses program.

A. Deklarasi Struktur dan Variabel

Struct RobuxItem digunakan untuk menyimpan informasi item Robux, yang terdiri dari:

nama: Nama item Robux.

harga: Harga per unit (tipe int).

stok: Jumlah stok tersedia.

Array robux[MAX_DATA] menyimpan semua item Robux dalam bentuk struct.

Variabel totalRobux menyimpan jumlah item yang tersedia.

Variabel login digunakan untuk autentikasi pengguna.

B. Sistem Login

Program meminta **username dan password**.

Jika salah, pengguna memiliki **3 kesempatan** untuk login.

Terdapat **dua jenis pengguna**:

Admin (admin, admin123): Bisa mengelola data.

User (febri, 2409106090): Hanya bisa melihat dan membeli.

Jika gagal login 3 kali, program keluar.

C. Menu Admin

1. Tambah Robux

Meminta input **nama, harga, dan stok**.

Menambah data ke dalam array robux.

Mengecek apakah array penuh sebelum menambah.

2. Tampilkan Robux

Menampilkan daftar item yang tersedia dalam format tabel.

3. Edit Robux

Memilih item berdasarkan nomor indeks.

Memungkinkan admin mengubah **nama, harga, dan stok**.

4. Hapus Robux

Memilih item berdasarkan nomor indeks.

Menggeser data untuk menghapus item.

5. Logout

Kembali ke halaman login.

6. Keluar Program

Menghentikan program.

D. Menu User

1. Lihat Robux

Sama seperti fitur admin, hanya bisa melihat daftar item.

2. Beli Robux

Memilih item berdasarkan nomor indeks.

Memasukkan jumlah yang ingin dibeli.

Jika stok mencukupi, stok akan dikurangi sesuai jumlah pembelian.

3. Logout dan Keluar

Keluar dari program.

3.source code

```
113
114 // ===== PENERAPAN POINTER =====
115 void tambahStokPointer(RobuxItem *data, int index, int tambah) {
116     if (index >= 0) {
117         data[index].stok += tambah;
118         cout << "Stok berhasil ditambahkan! Stok sekarang: " << data[index].stok << "\n";
119     } else {
120         cout << "Index tidak valid!\n";
121     }
122 }
123
124 void tampilkanDetailItem(RobuxItem *item) {
125     cout << "\n===== Detail Item Robux =====\n";
126     cout << "Nama : " << item->nama << "\n";
127     cout << "Harga : " << item->harga << "\n";
128     cout << "Stok : " << item->stok << "\n";
129     cout << "===== \n";
130 }
131
132 // ===== FUNGSI LOGIN =====
133 bool login(string user, string pass, bool &isAdmin) {
134     if (user == "admin" && pass == "admin123") {
135         isAdmin = true;
136         return true;
137     } else if (user == "febri" && pass == "2409106090") {
138         isAdmin = false;
139         return true;
140     }
141     return false;
142 }
143
144 // ===== MAIN PROGRAM =====
145 int main() {
146     string username, password;
147     // =====
```

1.Fungsi tambah stok pointer

1.1. Fungsi ini menerima pointer ke array of struct RobuxItem (RobuxItem *data).

1.2. Digunakan untuk menambahkan stok item Robux tertentu.

1.3. Pemanggilan dilakukan dengan address-of operator (&)

2.Fungsi tampilkan detail item

2.1. Fungsi ini menerima pointer ke satu buah RobuxItem, yaitu *item.

2.2. Di dalam fungsi, digunakan dereference operator (->) untuk mengakses atribut struct melalui pointer.