

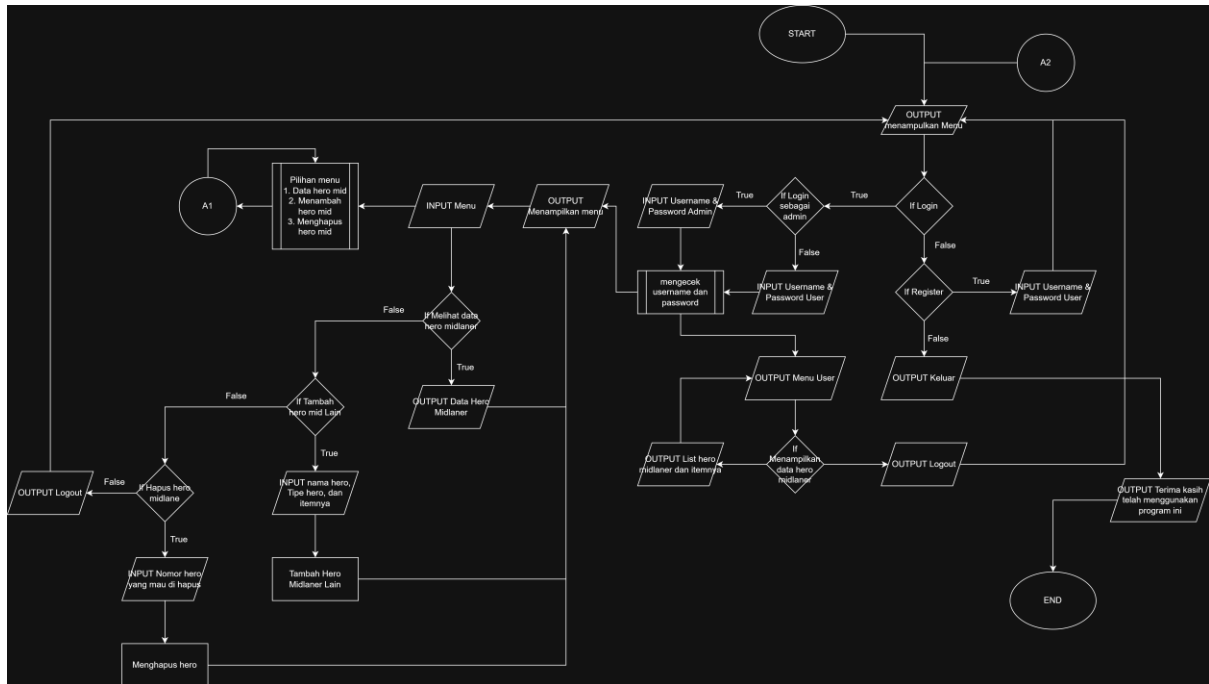
**LAPORAN PRAKTIKUM**  
**POSTTEST (7)**  
**ALGORITMA PEMROGRAMAN DASAR**

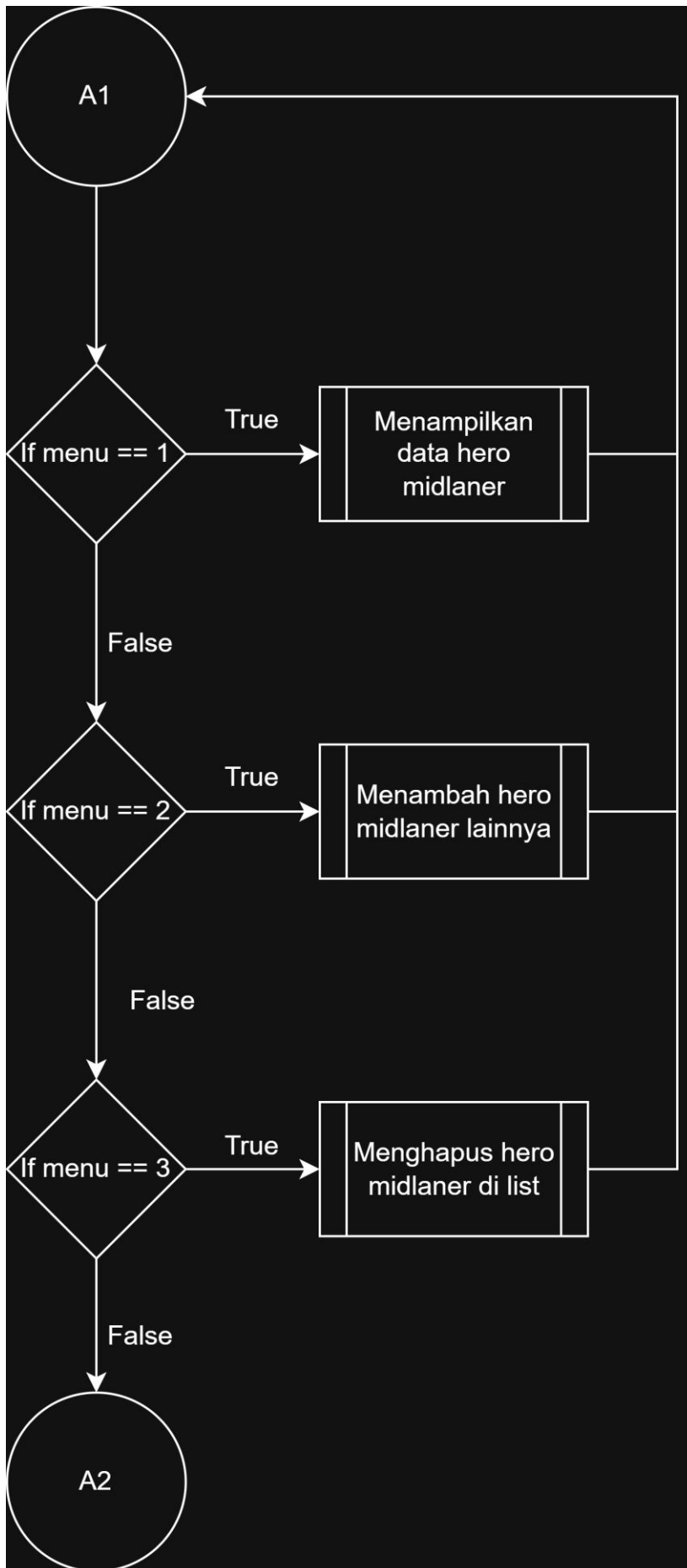


**Disusun oleh:**  
**Nama Bintang Alfianto Firdaus(2509106132)**  
**Kelas (C2 '25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

## 1. Flowchart





## 2. Deskripsi Singkat Program

Program untuk menampilkan list item mid laner dengan tipe data list dictionary dan function

## 3. Source Code

```
import os

# === VARIABEL GLOBAL ===
users = {
    1: {"username": "admin", "password": "123", "role": "admin"},
    2: {"username": "user", "password": "123", "role": "user"}
}

midlaners = {
    1: {"nama": "Lunox", "tipe": "Burst", "items": ["Genius Wand", "Glowing Wand", "Arcane Boots", "Lightning Truncheon", "Holy Crystal", "Blood Wings"]},
    2: {"nama": "Yve", "tipe": "Poke", "items": ["Enchanted Talisman", "Glowing Wand", "Demon Shoes", "Feather of Heaven", "Divine Glaive", "Blood Wings"]},
    3: {"nama": "Kagura", "tipe": "Burst", "items": ["Genius Wand", "Glowing Wand", "Arcane Boots", "Lightning Truncheon", "Holy Crystal", "Blood Wings"]}
}

user_aktif = None

# === FUNGSI TANPA PARAMETER ===
def clear_screen():
    """Membersihkan layar"""
    os.system('cls' if os.name == 'nt' else 'clear')

# === FUNGSI DENGAN PARAMETER ===
def login(username, password):
    """Mengembalikan data user jika login benar"""
    for id_user, data in users.items():
        if data["username"] == username and data["password"] == password:
            return {"id": id_user, **data}
    return None

# === FUNGSI TANPA PARAMETER ===
```

```

def tampilkan_midlaner():
    """Menampilkan seluruh data midlaner menggunakan fungsi rekursif"""
    def tampilkan_rekursif(index=0):
        if index >= len(midlaners):
            return
        idh = list(midlaners.keys())[index]
        data = midlaners[idh]
        print(f"{idh}\t{data['nama']:<10}\t{data['tipe']:<10}\t{'',
'.join(data['items'])}")
        tampilkan_rekursif(index + 1)

    print("ID\tNama\t\tTipe\t\tItem")
    print("-" * 60)
    tampilkan_rekursif()
    print("-" * 60)

# === FUNGSI DENGAN PARAMETER ===
def tambah_hero(nama, tipe, items):
    """Menambah hero baru ke dalam dictionary midlaners"""
    try:
        if nama == "" or tipe == "":
            raise ValueError("Nama dan tipe tidak boleh kosong!")

        new_id = max(midlaners.keys()) + 1 if midlaners else 1
        midlaners[new_id] = {"nama": nama, "tipe": tipe, "items": items}
        print("Hero berhasil ditambahkan!")
    except Exception as e:
        print(f"Terjadi kesalahan: {e}")

def menu_admin():
    """Menampilkan menu khusus admin"""
    global midlaners
    while True:
        clear_screen()
        print("=== MENU ADMIN ===")
        print("1. Lihat Hero Midlaner")
        print("2. Tambah Hero Lain")
        print("3. Hapus Hero")
        print("4. Logout")

        pilihan = input("Pilih menu: ")

        if pilihan == "1":
            clear_screen()
            tampilkan_midlaner()

```

```

        input("Tekan Enter untuk kembali...")

    elif pilihan == "2":
        clear_screen()
        print("=== TAMBAH HERO LAIN ===")
        nama = input("Nama hero: ")
        tipe = input("Tipe hero (Burst/Poke): ")
        items = []
        print("Masukkan maksimal 6 item (ketik '-' untuk berhenti):")

        while len(items) < 6:
            item = input(f"Item {len(items)+1}: ")
            if item == "-":
                break
            if item != "":
                items.append(item)
            else:
                print("Item tidak boleh kosong!")

        tambah_hero(nama, tipe, items)
        input("Tekan Enter untuk kembali...")

    elif pilihan == "3":
        clear_screen()
        print("=== HAPUS HERO ===")
        tampilkan_midlaner()

        try:
            id_hero = int(input("Masukkan ID Hero yang ingin dihapus:
"))

            if id_hero in midlaners:
                del midlaners[id_hero]
                print("Hero berhasil dihapus!")
            else:
                raise KeyError("ID tidak ditemukan!")
        except ValueError:
            print("Input harus berupa angka!")
        except KeyError as e:
            print(f"Error: {e}")
        finally:
            input("Tekan Enter untuk kembali...")

    elif pilihan == "4":
        break
    else:
        print("Pilihan tidak valid!")
        input("Tekan Enter untuk kembali...")

```

```

# === PROSEDUR 2 ===
def menu_user():
    """Menampilkan menu untuk user biasa"""
    while True:
        clear_screen()
        print("=== MENU USER ===")
        print("1. Lihat Hero Midlaner")
        print("2. Logout")

        pilih = input("Pilih menu: ")

        if pilih == "1":
            clear_screen()
            tampilkan_midlaner()
            input("Tekan Enter untuk kembali...")
        elif pilih == "2":
            break
        else:
            print("Pilihan tidak valid!")
            input("Tekan Enter untuk kembali...")

# === PROGRAM UTAMA ===
while True:
    clear_screen()
    print("=== LIST ITEM MIDLANER MOBILE LEGENDS ===")
    print("1. Login")
    print("2. Register")
    print("3. Keluar")

    menu = input("Pilih menu: ")

    if menu == "1":
        clear_screen()
        print("=== LOGIN ===")
        username = input("Username: ")
        password = input("Password: ")

        user_aktif = login(username, password)

        if user_aktif is None:
            print("Username atau password salah!")
            input("Tekan Enter untuk kembali...")
            continue

```

```

        print(f"Login berhasil! Selamat datang,
{user_aktif['role'].capitalize()} {user_aktif['username']}!")
        input("Tekan Enter untuk melanjutkan...")

    if user_aktif["role"] == "admin":
        menu_admin()
    else:
        menu_user()

elif menu == "2":
    clear_screen()
    print("=== REGISTER ===")
    username = input("Username baru: ")
    password = input("Password baru: ")

    try:
        if username == "" or password == "":
            raise ValueError("Username dan password tidak boleh
kosong!")

        if any(u["username"] == username for u in users.values()):
            raise ValueError("Username sudah digunakan!")

        new_id = max(users.keys()) + 1 if users else 1
        users[new_id] = {"username": username, "password": password,
"role": "user"}
        print("Akun berhasil dibuat!")
    except ValueError as e:
        print(f"Error: {e}")
    finally:
        input("Tekan Enter untuk kembali...")

elif menu == "3":
    clear_screen()
    print("Terima kasih telah menggunakan program ini!")
    break

else:
    print("Pilihan tidak valid!")
    input("Tekan Enter untuk kembali...")

```



## 4. Hasil Output

(Sertakan tangkapan layar atau hasil output dari program setelah dijalankan.)

ID	Nama	Tipe	Item
1	Lunox	Burst	Genius Wand, Glowing Wand, Arcane Boots, Lightning Truncheon, Holy Crystal, Blood Wings
2	Yve	Poke	Enchanted Talisman, Glowing Wand, Demon Shoes, Feather of Heaven, Divine Glaive, Blood Wings
3	Kagura	Burst	Genius Wand, Glowing Wand, Arcane Boots, Lightning Truncheon, Holy Crystal, Blood Wings

Tekan Enter untuk kembali...

Gambar 4.1 Contoh

Gambar <nomor bab>.[index] <judul-gambar>

## 5. Langkah-langkah GIT

(Berikan screenshot dan jelaskan secara ringkas fungsi dari yang kalian ketik)

### 5.1 GIT Add

```
$ git add .
```

### 5.2 GIT Commit

```
$ git commit -m "first commit"
On branch master

Initial commit

nothing to commit (create/copy files and use "git add" to track)
```

### 5.3 GIT Push

```
$ git push -u origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 89.77 KiB | 11.22 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ezaayt07-prog/pertemuan-1.git
   b144d63..03cad72  main -> main
branch 'main' set up to track 'origin/main'.
```