

LAPORAN PRAKTIKUM
POSTTEST (6)
ALGORITMA PEMROGRAMAN DASAR



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

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

```

graph TD
    START([START]) --> Menu[/Menu/]
    Menu --> IfRegister{If register}
    IfRegister -- True --> InputUsername[/INPUT username dan password/]
    IfRegister -- False --> Logout[/OUTPUT logout/]
    InputUsername --> CheckInput1[Memeriksa input benar menggunakan data]
    CheckInput1 --> IfLogin1{If Login}
    IfLogin1 -- True --> Welcome[/OUTPUT Selamat, aku anda telah di buat/]
    IfLogin1 -- False --> BackMenu1[/OUTPUT kembali ke menu/]
    Welcome --> IfUser{If user}
    IfUser -- True --> IfRegistration{If registrasi berhasil}
    IfUser -- False --> BackMenu1
    IfRegistration -- True --> AccountExists[/OUTPUT akun sudah ada/]
    IfRegistration -- False --> LoginSuccess[/OUTPUT login berhasil/]
    LoginSuccess --> IfAdmin{If Admin}
    IfAdmin -- True --> LoginSuccess
    IfAdmin -- False --> ManageList[/OUTPUT dapat mengatur list/]
    ManageList --> DeleteList[/OUTPUT dapat menghapus list/]
    DeleteList --> DisplayList[/OUTPUT menampilkan daftar hero dan itemnya/]
    DisplayList --> IfAdminLogin{If login sebagai admin}
    IfAdminLogin -- True --> EditList[/OUTPUT edit list/]
    IfAdminLogin -- False --> END([END])
  
```

2. Deskripsi Singkat Program

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

3. Source Code

```
import os

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"]}
}

while True:
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== LIST ITEM MIDLANER MOBILE LEGENDS ===")
    print("1. Login")
    print("2. Register")
    print("3. Keluar")

    menu = input("Pilih menu: ")

    # === LOGIN ===
    if menu == "1":
        os.system('cls' if os.name == 'nt' else 'clear')
        print("=== LOGIN ===")
        username = input("Username: ")
        password = input("Password: ")

        user_aktif = None
        for id_user, data in users.items():
            if data["username"] == username and data["password"] == password:
```

```

        user_aktif = {"id": id_user, **data}
        break

    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...")

# === MENU ADMIN ===
if user_aktif["role"] == "admin":
    while True:
        os.system('cls' if os.name == 'nt' else 'clear')
        print("=== MENU ADMIN ===")
        print("1. Lihat Hero Midlaner")
        print("2. Tambah Hero Baru")
        print("3. Update Hero")
        print("4. Hapus Hero")
        print("5. Logout")

        pilihan = input("Pilih menu: ")

        # Lihat hero
        if pilihan == "1":
            os.system('cls' if os.name == 'nt' else 'clear')
            print("=== DAFTAR HERO MIDLANER ===")
            print("ID\tNama\t\tTipe\t\tItem")
            print("-" * 60)
            for idh, data in midlaners.items():
                print(f"{idh}\t{data['nama']:<10}\t{data['tipe']:<10}
}\t{'', '.join(data['items'])}")
            print("-" * 60)
            input("Tekan Enter untuk kembali...")

        # Tambah hero
        elif pilihan == "2":
            os.system('cls' if os.name == 'nt' else 'clear')
            print("=== TAMBAH HERO BARU ===")
            nama = input("Nama hero: ")
            tipe = input("Tipe hero (Burst/Poke): ")

            if nama == "" or tipe == "":
                print("Nama dan tipe tidak boleh kosong!")
                input("Tekan Enter untuk kembali...")
                continue

```

```

        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("Nama item 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!")
        input("Tekan Enter untuk kembali...")

    # Update hero
    elif pilihan == "3":
        os.system('cls' if os.name == 'nt' else 'clear')
        print("=== UPDATE HERO ===")
        print("ID\tNama\t\tTipe")
        print("-" * 40)
        for idh, data in midlaners.items():
            print(f"{idh}\t{data['nama']:<10}\t{data['tipe']}")
        print("-" * 40)

        id_hero = input("Masukkan ID Hero yang ingin diupdate:
")

        if not id_hero.isdigit():
            print("ID harus berupa angka!")
            input("Tekan Enter untuk kembali...")
            continue

        id_hero = int(id_hero)
        if id_hero not in midlaners:
            print("ID Hero tidak ditemukan!")
            input("Tekan Enter untuk kembali...")
            continue

        data = midlaners[id_hero]
        print(f"Update hero: {data['nama']}")
        nama_baru = input("Nama baru (kosongkan jika tidak
diubah): ")

        tipe_baru = input("Tipe baru (kosongkan jika tidak
diubah): ")

```

```

        if nama_baru != "":
            data["nama"] = nama_baru
        if tipe_baru != "":
            data["tipe"] = tipe_baru

        ubah_item = input("Ingin ubah item? (y/n): ")
        if ubah_item.lower() == "y":
            items_baru = []
            print("Masukkan item baru (maks 6, '-' untuk
berhenti):")

            while len(items_baru) < 6:
                item = input(f"Item {len(items_baru)+1}: ")
                if item == "-":
                    break
                if item != "":
                    items_baru.append(item)
            if items_baru:
                data["items"] = items_baru

        print("Data hero berhasil diperbarui!")
        input("Tekan Enter untuk kembali...")

# Hapus hero
elif pilihan == "4":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== HAPUS HERO ===")
    print("ID\tNama\t\tTipe")
    print("-" * 40)
    for idh, data in midlaners.items():
        print(f"{idh}\t{data['nama']:<10}\t{data['tipe']}")
    print("-" * 40)

    id_hero = input("Masukkan ID Hero yang ingin dihapus: ")
    if not id_hero.isdigit():
        print("ID harus berupa angka!")
        input("Tekan Enter untuk kembali...")
        continue

    id_hero = int(id_hero)
    if id_hero in midlaners:
        del midlaners[id_hero]
        print("Hero berhasil dihapus!")
    else:
        print("ID Hero tidak ditemukan!")
        input("Tekan Enter untuk kembali...")

elif pilihan == "5":

```

```

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

# === MENU USER ===
else:
    while True:
        os.system('cls' if os.name == 'nt' else 'clear')
        print("=== MENU USER ===")
        print("1. Lihat Hero Midlaner")
        print("2. Logout")

        pilih = input("Pilih menu: ")

        if pilih == "1":
            os.system('cls' if os.name == 'nt' else 'clear')
            print("=== DAFTAR HERO MIDLANER ===")
            print("ID\tNama\t\tTipe\t\tItem")
            print("-" * 60)
            for idh, data in midlaners.items():
                print(f"{idh}\t{data['nama']:<10}\t{data['tipe']:<10}
}\t{'', '.join(data['items'])}")
            print("-" * 60)
            input("Tekan Enter untuk kembali...")
        elif pilih == "2":
            break
        else:
            print("Pilihan tidak valid!")
            input("Tekan Enter untuk kembali...")

# === REGISTER ===
elif menu == "2":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("=== REGISTER AKUN BARU ===")
    username = input("Masukkan username baru: ")
    password = input("Masukkan password: ")

    if username == "" or password == "":
        print("Username dan password tidak boleh kosong!")
        input("Tekan Enter untuk kembali...")
        continue

    sudah_ada = any(data["username"] == username for data in
users.values())

    if sudah_ada:
        print("Username sudah digunakan!")

```

```

        else:
            new_id = max(users.keys()) + 1 if users else 1
            users[new_id] = {"username": username, "password": password,
"role": "user"}
            print("Akun berhasil dibuat!")

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

# === KELUAR ===
elif menu == "3":
    os.system('cls' if os.name == 'nt' else 'clear')
    print("Terima kasih telah menggunakan program List Item Midlaner
ML!")
    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'.
```