

LAPORAN PRAKTIKUM
POSTTEST 7
ALGORITMA PEMROGRAMAN DASAR

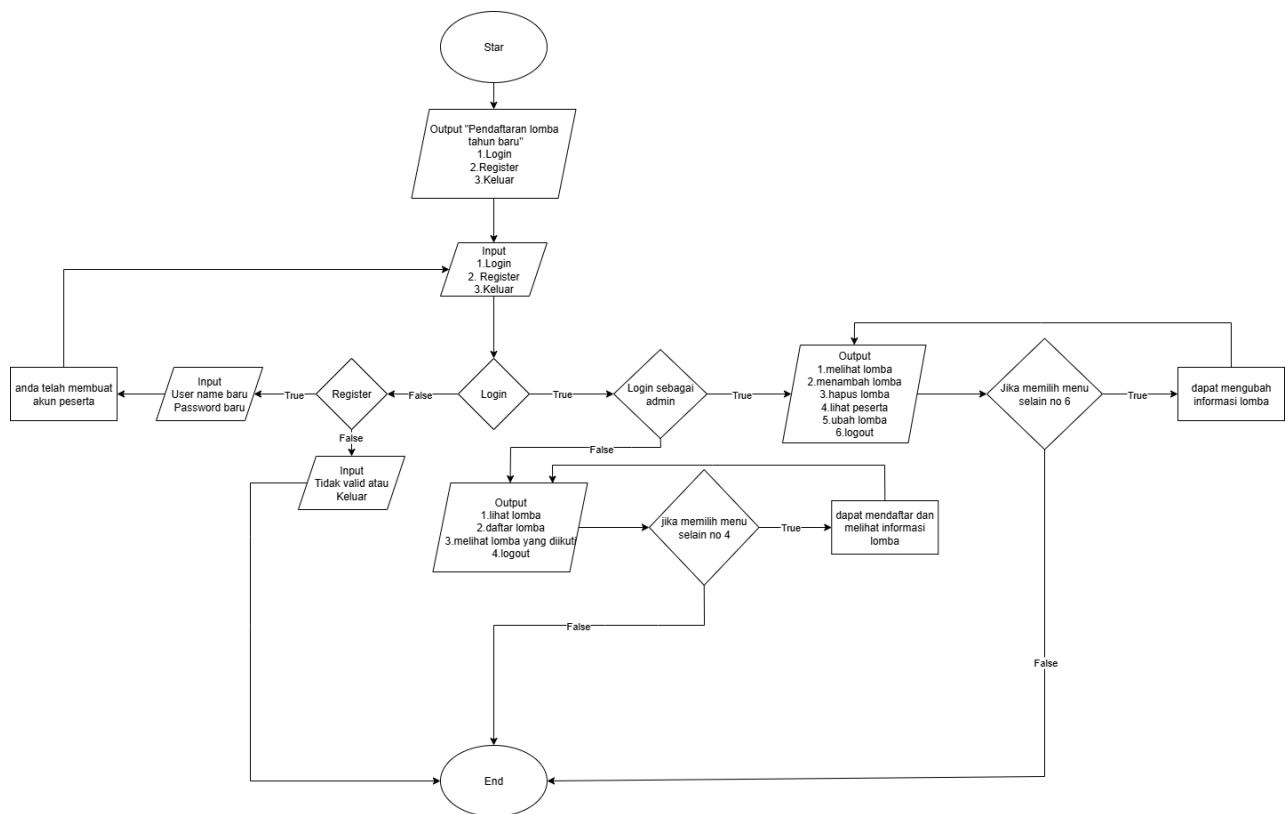


Disusun oleh:
ALBERT EINSTEIN LIEM (2509106095)
Kelas (C1 '25)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart (CRUD) Pendaftaran Lomba Tahun Baru

Pertama kita masukan *Output* lalu *Input* untuk “Login, Register ,dan Keluar” kemudian kita tambahkan *Decision*, di sini kita hanya mengikuti apa yang ada di *Input* sebelum nya seperti jika dia *False* yang berarti *Register* di bagian *True* maka ia akan membuat akun baru sebagai peserta lomba, kemudian di bagian *False* ia akan otomatis keluar program karna “tidak valid” atau “keluar”, nah setelah program nya selesai membuat akun baru ia akan kembali ke input pertama, di bagian *Decision*, jika *True* program akan masuk ke settingan Admin, *False* program akan lanjut ke menu Peserta, jika Admin ia dapat mengubah semua informasi yang ada di dalam lomba, dan jika login sebagai peserta ia hanya dapat mendaftar dan melihat lomba yang ada.



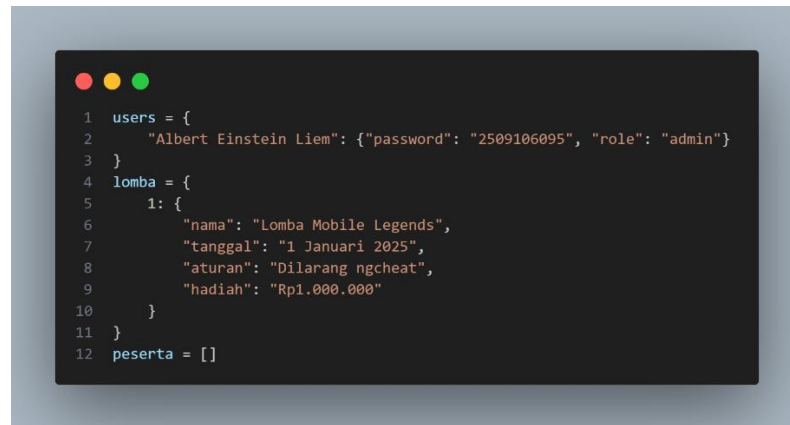
Gambar 1.1 Flowchart

2. Deskripsi Singkat Program

Program ini merupakan simulasi pendaftaran lomba tahun baru secara online, dengan Admin yang dapat mengubah info lomba dan dengan peserta yang dapat mendaftar lomba, program ini juga dapat mempermudah masyarakat untuk mengetahui lomba tahun baru apa saja yang sedang di laksanakan atau yang akan di laksanakan, program ini juga dapat mempermudah Masyarakat untuk mendaftar lomba secara online.

3. Source Code

A. Variabel Global



```
1 users = {
2     "Albert Einstein Liem": {"password": "2509106095", "role": "admin"}
3 }
4 lomba = {
5     1: {
6         "nama": "Lomba Mobile Legends",
7         "tanggal": "1 Januari 2025",
8         "aturan": "Dilarang ngcheat",
9         "hadiah": "Rp1.000.000"
10    }
11 }
12 peserta = []
```

Gambar 3.1 Python Variabel Global

B. Parameter 1



```
1 def tampil_lomba(data_lomba):
2     if len(data_lomba) == 0:
3         print("Belum ada lomba terdaftar.")
4     else:
5         for i, d in data_lomba.items():
6             print(f"{i}. {d['nama']}")
7             print(f"    Tanggal : {d['tanggal']}")
8             print(f"    Peraturan: {d['aturan']}")
9             print(f"    Hadiah   : {d['hadiah']}")
10            print()
```

Gambar 3.2 Parameter 1

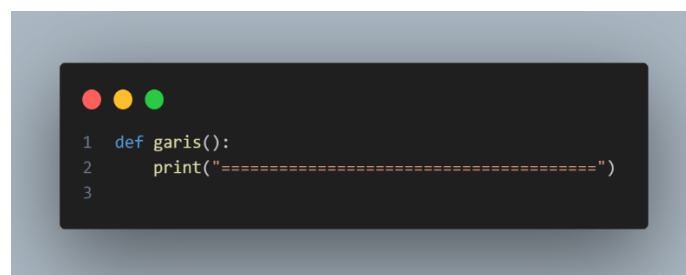
C. Parameter 2

A screenshot of a Python code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code defines a function named `valid_angka` that takes a parameter `teks`.

```
1 def valid_angka(teks):  
2     try:  
3         int(teks)  
4         return True  
5     except ValueError:  
6         return False
```

Gambar 3.3 Parameter 2

D. Tanpa Parameter 1

A screenshot of a Python code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code defines a function named `garis` that does not take any parameters.

```
1 def garis():  
2     print("=====")  
3
```

Gambar 3.4 Tanpa Parameter 1

E. Tanpa Parameter 2

A screenshot of a Python code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code defines a function named `pause` that does not take any parameters.

```
1 def pause():  
2     input("Pencet enter untuk lanjut...")  
3
```

Gambar 3.5 Tanpa Parameter 2

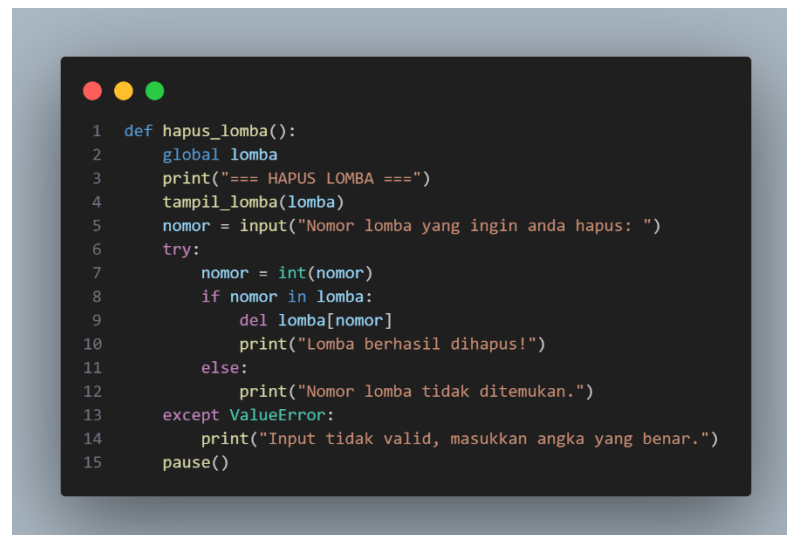
F. Prosedur 1



```
1 def tambah_lomba():
2     global lomba
3     print("=== TAMBAH LOMBA BARU ===")
4     nama_lomba = input("Lomba: ")
5     hari = input("Tanggal lomba: ")
6     aturan = input("Aturan lomba: ")
7     hadiah = input("Hadiah lomba: ")
8
9     try:
10        id_baru = len(lomba) + 1
11        lomba[id_baru] = {}
12        lomba[id_baru]["nama"] = nama_lomba
13        lomba[id_baru]["tanggal"] = hari
14        lomba[id_baru]["aturan"] = aturan
15        lomba[id_baru]["hadiah"] = hadiah
16        print("Lomba berhasil ditambahkan!")
17    except ValueError:
18        print("Ada kesalahan ketika menambah lomba.")
19    pause()
```

Gambar 3.6 Prosedur 1

G. Prosedur 2



```
1 def hapus_lomba():
2     global lomba
3     print("=== HAPUS LOMBA ===")
4     tampil_lomba(lomba)
5     nomor = input("Nomor lomba yang ingin anda hapus: ")
6     try:
7         nomor = int(nomor)
8         if nomor in lomba:
9             del lomba[nomor]
10            print("Lomba berhasil dihapus!")
11        else:
12            print("Nomor lomba tidak ditemukan.")
13    except ValueError:
14        print("Input tidak valid, masukkan angka yang benar.")
15    pause()
```

Gambar 3.7 Prosedur 1

H. Rekursif Untuk Menu Utama

A screenshot of a code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. The code is written in Python and defines a recursive function named `menu_utama()`. The function prints a title, a separator line, and three menu options: "1. Login", "2. Register", and "3. Keluar". It then prompts the user to "Pilih menu: ". Based on the input, it either calls `login()`, `register()`, or prints a thank-you message and returns. For invalid input, it prints "Pilihan tidak valid!", calls `pause()`, and recursively calls `menu_utama()` again.

```
1 def menu_utama():
2     garis()
3     print("=== DAFTAR LOMBA TAHUN BARU ===")
4     garis()
5     print("1. Login")
6     print("2. Register")
7     print("3. Keluar")
8     garis()
9     menu = input("Pilih menu: ")
10
11     if menu == "1":
12         login()
13     elif menu == "2":
14         register()
15     elif menu == "3":
16         print("Terima kasih! Semoga beruntung!!")
17         return
18     else:
19         print("Pilihan tidak valid!")
20         pause()
21         menu_utama()
```

Gambar 3.8 Python Rekursif Menu Utama

4. Hasil Output

```
=== PENDAFTARAN LOMBA TAHUN BARU ===  
=====
```

1. Login
2. Register
3. Keluar

```
=====
```

pilih menu : █

Gambar 4.1 menu pilihan login

```
=== LOGIN AKUN ===  
  
Username: Albert Einstein Liem  
Password: 2509106095█
```

Gambar 4.2 login sebagai admin

```
=== MENU ADMIN ===  
  
1. Lihat Semua Lomba  
2. Tambah Lomba  
3. Ubah Lomba  
4. Hapus Lomba  
5. Lihat Peserta  
6. Logout  
  
Pilih menu: █
```

Gambar 4.3 menu admin

```
=== DAFTAR LOMBA ===  
1 . Turnament Mobile Legend  
Tanggal Mulai : 1 Januari 2025  
    Peraturan   : Tidak boleh cheat  
    Hadiah      : Hadiah uang Rp1.000.000  
  
Tekan Enter untuk melanjutkan...█
```

Gambar 4.4 melihat lomba

```
=== TAMBAH LOMBA BARU ===  
Lomba: lari  
Hari lomba: 2 januari 2025  
Peraturan lomba: tidak boleh lewat jalan pintas  
Hadiah lomba: Rp1.000.000  
Lomba berhasil ditambahkan!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.5 tambah lomba

```
=== UBAH DATA LOMBA ===  
1 . Turnament Mobile Legend  
2 . lari  
Pilih no lomba yang ingin di ubah: 2  
Data sebelumnya:  
Nama      : lari  
Tanggal   : 2 januari 2025  
Peraturan: tidak boleh lewat jalan pintas  
Hadiah    : Rp1.000.000  
Masukkan data baru  
Lomba baru: locan jauh  
Tanggal mulai lomba baru: 2 januari 2025  
Peraturan baru: tidak boleh melewati garis yang sudah di tentukan  
Hadiah baru: Rp1.000.000  
Data lomba berhasil diubah!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.6 perubahan lomba

```
=== HAPUS LOMBA ===  
1. Turnament Mobile Legend  
2. locan jauh  
Nomor lomba yang mau dihapus: 2  
Lomba berhasil dihapus!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.7 hapus lomba

```
=== DAFTAR PESERTA ===  
ipin - Turnament Mobile Legend  
Tekan Enter untuk melanjutkan...
```

Gambar 4.8 melihat daftar peserta


```
=== REGISTER AKUN BARU ===
```

```
Masukkan username baru: ipin  
Masukkan password baru: ipin1  
Akun lomba anda sudah dibuat  
Tekan Enter untuk melanjutkan...
```

Gambar 4.9 pembuatan akun peserta

```
=== LOGIN AKUN ===
```

```
Username: ipin  
Password: ipin1
```

Gambar 4.10 login sebagai peserta

```
=== MENU PENGGUNA ===
```

```
1. Lihat Lomba  
2. Daftar Lomba  
3. Lihat Lomba yang Diikuti  
4. Logout  
  
Pilih menu: 
```

Gambar 4.11 menu peserta

```
=== DAFTAR KE LOMBA ===
```

```
1. Turnament Mobile Legend  
Pilih nomor lomba yang kamu minati: 1  
Berhasil mendaftar ke Turnament Mobile Legend!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.12 daftar lomba

```
Pilih menu: 2  
1. Lomba Mobile Legends  
   Tanggal   : 1 Januari 2025  
   Peraturan : Dilarang ngcheat  
   Hadiah    : Rp1.000.000  
  
Pilih nomor lomba: d  
Input harus berupa angka.
```

Gambar 4.13 Error Handling

5. Langkah-langkah GIT

5.1 GIT Add

memilih file yang mau dicatat sebelum disimpan.

```
PS C:\Kenzyy\kuliah\github\Praktikum-apd> git add .
```

5.2 GIT Commit

menyimpan perubahan itu secara permanen di *repository lokal* (di komputer kita), lengkap dengan pesan.

```
PS C:\Kenzyy\kuliah\github\Praktikum-apd> git commit -m "post-test-7"
[main ad4e105] post-test-7
 4 files changed, 434 insertions(+), 1 deletion(-)
 create mode 100644 Praktikum-apd/kelas/pertemuan-7/pertemuan-7-error handling
 create mode 100644 Praktikum-apd/kelas/pertemuan-7/pertemuan-7-fungsi dan prosedur
 create mode 100644 Praktikum-apd/post-test/post-test-apd-7/2509106095-AlbertEinsteinLiem-PT-7.PY
```

5.3 GIT Push

mengirim commit dari komputer kita ke *repository GitHub* supaya tersimpan online dan bisa dilihat orang lain.

```
PS C:\Kenzyy\kuliah\github\Praktikum-apd> git push -u origin main
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 16 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (12/12), 3.40 KiB | 3.40 MiB/s, done.
Total 12 (delta 5), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (5/5), completed with 4 local objects.
To https://github.com/alberteinsteinliem-dev/praktikum-apd.git
   9b0d890..ad4e105  main -> main
branch 'main' set up to track 'origin/main'.
```