

LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN DASAR

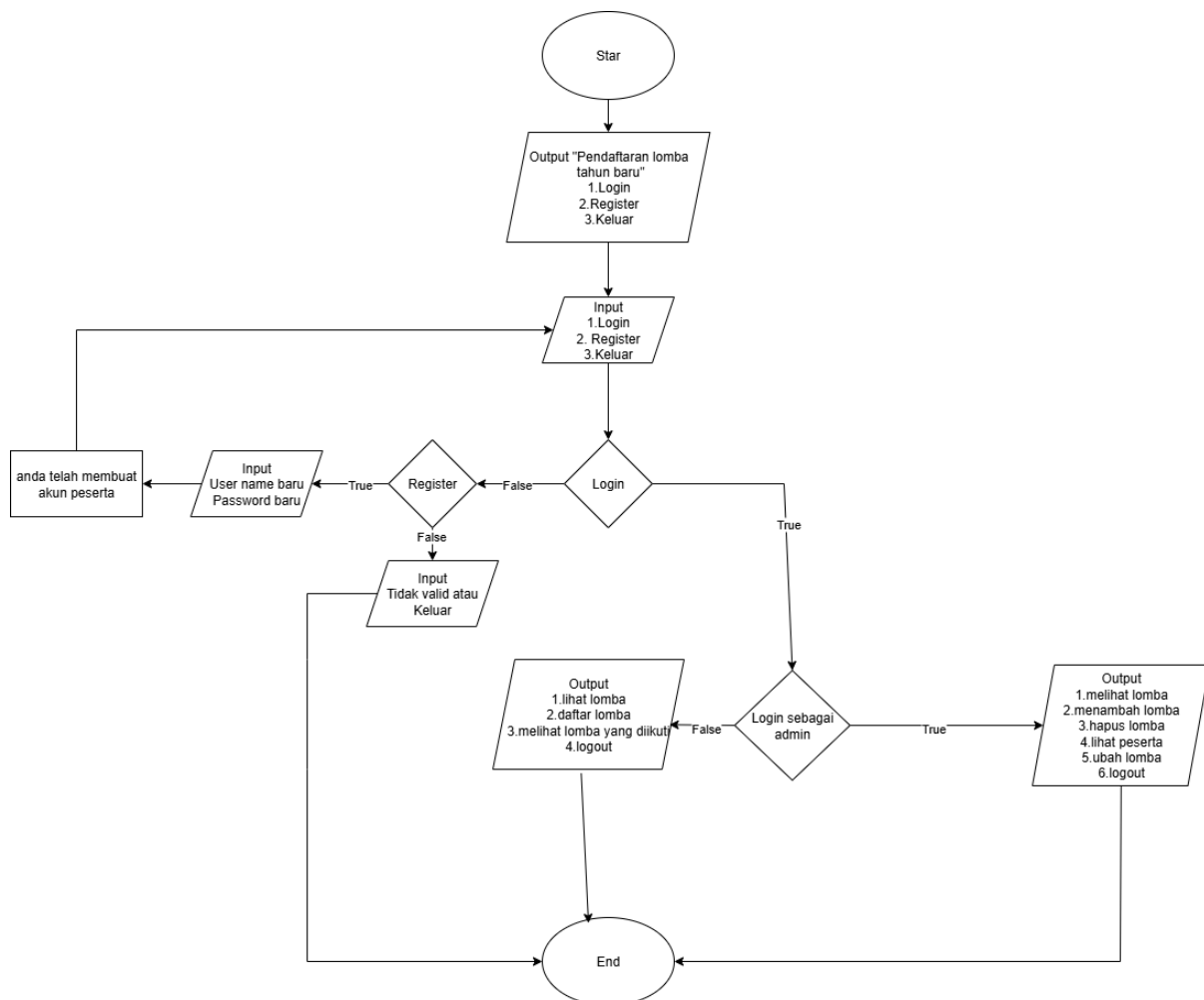


Disusun oleh:
ALBERT EINSTEIN LIEM (2509106095)
Kelas (C1 '25)

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. Fitur Login

A screenshot of a Python code editor with a dark background and light-colored text. The code is for a login feature. It starts with a while loop that runs indefinitely. Inside the loop, it clears the terminal, prints a menu, and asks the user to choose a menu. If the user chooses '1', it enters another while loop for the login process, where it asks for a username and password.

```
1 while True:
2     os.system(("cls", "clear")[os.name != "nt"])
3     print("=== DAFTAR LOMBA TAHUN BARU ===")
4     print("=====")
5     print("1. Login")
6     print("2. Register")
7     print("3. Keluar")
8     print("=====")
9     menu = input("Pilih menu: ")
10
11     if menu == "1":
12         os.system(("cls", "clear")[os.name != "nt"])
13         print("=== LOGIN ===")
14         nama = input("Username: ")
15         pw = input("Password: ")
16
```

Gambar 2.1 Python Login

B. Fitur Admin

A screenshot of a Python code editor with a dark background and light-colored text. The code is for an admin feature. It starts with an if statement that checks if a user is in the 'users' dictionary and if their password matches. If both conditions are met and the user's role is 'admin', it enters a while loop that runs indefinitely. Inside the loop, it clears the terminal, prints an admin menu, and asks the user to choose a menu. The menu options include viewing all competitions, adding, changing, deleting competitions, viewing participants, and logging out.

```
1 if nama in users and users[nama]["password"] == pw:
2     role = users[nama]["role"]
3     if role == "admin":
4         while True:
5             os.system(("cls", "clear")[os.name != "nt"])
6             print("=== MENU ADMIN ===")
7             print("1. Lihat Semua Lomba")
8             print("2. Tambah Lomba")
9             print("3. Ubah Lomba")
10            print("4. Hapus Lomba")
11            print("5. Lihat Peserta")
12            print("6. Logout")
13            pilih = input("Pilih menu: ")
14
```

Gambar 3.1 python Fitur Admin

C. Program Yang Dapat di Ubah Oleh Admin

```
1  if pilih == "1":
2      os.system(("cls", "clear"))[os.name != "nt"])
3      print("=== DAFTAR LOMBA ===")
4      if len(lomba) == 0:
5          print("Belum ada lomba terdaftar.")
6      else:
7          for i, data in lomba.items():
8              print(i, ".", data["nama"])
9              print(f"    Tanggal : {data['tanggal']}")
10             print(f"    Peraturan: {data['aturan']}")
11             print(f"    Hadiah   : {data['hadiah']}")
12             print()
13             input("Pencet enter untuk lanjut")
14
15  elif pilih == "2":
16      os.system(("cls", "clear"))[os.name != "nt"])
17      print("=== TAMBAH LOMBA BARU ===")
18      nama_lomba = input("Lomba: ")
19      hari = input("Hari lomba: ")
20      peraturan = input("Peraturan lomba: ")
21      hadiah = input("Hadiah lomba: ")
22      id_baru = len(lomba) + 1
23      data_lomba = {
24          "nama": nama_lomba,
25          "tanggal": hari,
26          "aturan": peraturan,
27          "hadiah": hadiah
28      }
29      lomba[id_baru] = data_lomba
30      print("Lomba berhasil ditambahkan!")
31      input("Pencet enter untuk lanjut")
32
33  elif pilih == "3":
34      os.system(("cls", "clear"))[os.name != "nt"])
35      print("=== UBAH DATA LOMBA ===")
36      for i, data in lomba.items():
37          print(i, ".", data["nama"])
38          nomor = input("Nomor lomba yang mau diubah: ")
39          angka = True
40          for c in nomor:
41              angka = c in "0123456789"
42          if not angka:
43              angka = False
44          if angka and nomor != "":
45              nomor = int(nomor)
46              if nomor in lomba:
47                  print("Masukkan data lomba baru:")
48                  nama_baru = input("Nama : ")
49                  tanggal_baru = input("Tanggal: ")
50                  aturan_baru = input("Peraturan: ")
51                  hadiah_baru = input("Hadiah: ")
```

Gambar 3.2 python Program Admin

D. Fitur Peserta

```
1 else:
2     while True:
3         os.system(("cls", "clear")[os.name != "nt"])
4         print("=== MENU PENGGUNA ===")
5         print("1. Lihat Lomba")
6         print("2. Daftar Lomba")
7         print("3. Lihat Lomba yang Diikuti")
8         print("4. Logout")
9         pilih = input("Pilih menu: ")
10
```

Gambar 4.1 python Fitur Peserta

E. Fitur Yang Dapat di Akses Oleh Peserta

```
1 if pilih == "1":
2     os.system(("cls", "clear")[os.name != "nt"])
3     print("=== DAFTAR LOMBA ===")
4     for i, data in lomba.items():
5         print(i, ".", data["nama"])
6         print(f"    Tanggal : {data['tanggal']}")
7         print(f"    Peraturan: {data['aturan']}")
8         print(f"    Hadiah   : {data['hadiah']}")
9         print()
10    input("Pencet enter untuk lanjut")
11
12 elif pilih == "2":
13     os.system(("cls", "clear")[os.name != "nt"])
14     print("=== DAFTAR KE LOMBA ===")
15     for i, data in lomba.items():
16         print(i, ".", data["nama"])
17     nomor = input("Pilih nomor lomba yang kamu diminati: ")
18     angka = True
19     for c in nomor:
20         angka = c in "0123456789"
21         if not angka:
22             angka = False
23     if angka and nomor != "":
24         nomor = int(nomor)
25         if nomor in lomba:
26             peserta.append({"nama": nama, "lomba": lomba[nomor]["nama"]})
27             print(f"Berhasil mendaftar ke {lomba[nomor]['nama']}!")
28         else:
29             print("Nomor lomba tidak valid.")
30     else:
31         print("Input tidak valid.")
32     input("Pencet enter untuk lanjut")
33
34 elif pilih == "3":
35     os.system(("cls", "clear")[os.name != "nt"])
36     print("=== LOMBA YANG DIKUTI ===")
37     ada = False
38     for p in peserta:
39         if p["nama"] == nama:
40             print(".", p["lomba"])
41             ada = True
42     if not ada:
43         print("Belum daftar lomba apapun.")
44     input("Pencet enter untuk lanjut")
45
46 elif pilih == "4":
47     break
48
49 else:
50     print("Pilihan tidak valid!")
51     input("Pencet enter untuk lanjut")
52
53 else:
54     print("Login gagal! Username atau password salah.")
55     input("Pencet enter untuk lanjut")
56
```

Gambar 4.2 python Fitur Peserta

4. Hasil Output

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

1. Login
2. Register
3. Keluar

```
=====
```

pilih menu : █

Gambar 5.2 menu pilihan login

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

Gambar 5.2 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 6.1 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 6.2 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 6.3 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 7.1 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 7.2 hapus lomba

```

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

```

Gambar 7.3 melihat daftar peserta

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

Gambar 7.4 pembuatan akun peserta

```
=== LOGIN AKUN ===  
  
Username: ipin  
Password: ipin1
```

Gambar 8.1 login sebagai peserta

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

Gambar 8.2 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 8.3 daftar lomba

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-6"
[main 03e007a] post-test-6
2 files changed, 347 insertions(+)
create mode 100644 Praktikum-apd/kelas/pertemuan-6/pertemuan-6-set&dictionary
create mode 100644 Praktikum-apd/post-test/post-test-apd-6/2509106095-AlbertEinsteinLiem-PT-6.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: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 16 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 3.21 KiB | 3.21 MiB/s, done.
Total 9 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/alberteinsteinliem-dev/praktikum-apd.git
 e276764..03e007a main -> main
branch 'main' set up to track 'origin/main'.
```