

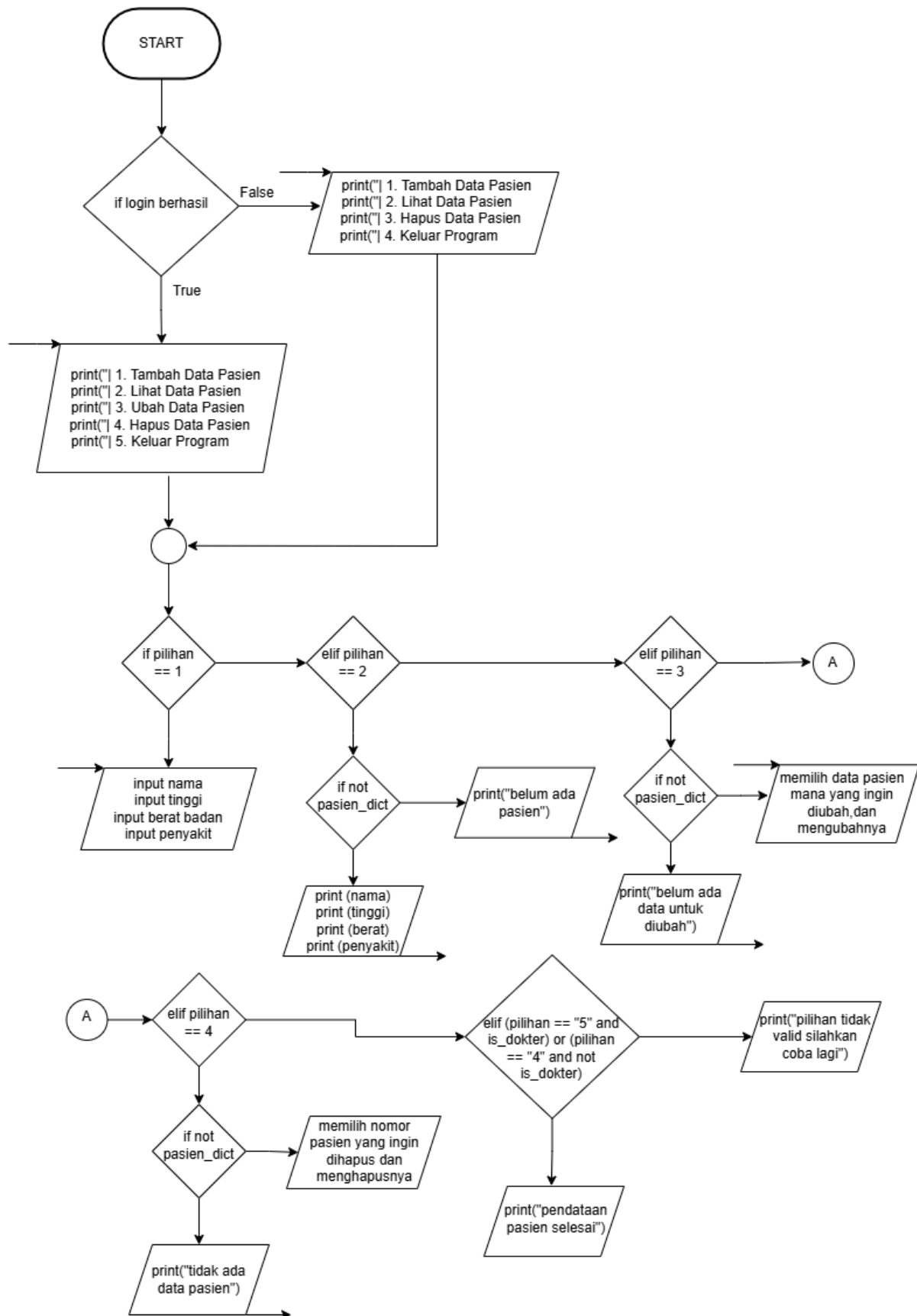
LAPORAN PRAKTIKUM
POSTTEST 6
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Muhammad Fakhri Nabal
2509106009
Kelas A1 '25

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1.Flowchart



Pertama program akan menampilkan halaman login pada user, jika user memasukkan

username dan password yang benar maka program akan menampilkan halaman menu pilihan dari 1 sampai 5. jika user tidak memasukkan username dan password dengan benar maka program akan menampilkan halaman menu pilihan dari 1 sampai 4 saja.

Jika user memilih nomor 1 maka program akan menampilkan halaman pendataan pasien baru. Jika user memilih nomor 2 maka program akan menampilkan halaman data pasien yang sudah ada yang hanya bisa dibaca saja (read only), jika tidak ada data pasien program akan mengeprint “belum ada pasien”. Jika user memilih nomor 3 maka program akan menampilkan halaman daftar nama pasien yang ada dan bisa dipilih untuk mengubah data.

Untuk pilihan nomor 4 ada 2 kondisi yang pertama jika user berhasil login maka pilihan nomor 4 akan menampilkan halaman data pasien yang sudah ada untuk dihapus. Jika user tidak berhasil login maka pilihan nomor 4 akan menampilkan keluar dari program *pendataan pasien*. pilihan nomor 5 akan mengeluarkan user dari program *pendataan pasien*, pilihan nomor 5 hanya muncul jika user berhasil login.

2. Deskripsi Singkat Program

Fungsi dari program ini adalah untuk memudahkan para tenaga medis untuk mendata setiap pasien yang ada.

3. Source Code

```
1  while True:
2      print("\n=====")
3      print("|          PENDATAAN PASIEN (CRUD)          |")
4      print("=====")
5
6      if is_dokter:
7          print("| 1. Tambah Data Pasien          |")
8          print("| 2. Lihat Data Pasien           |")
9          print("| 3. Ubah Data Pasien            |")
10         print("| 4. Hapus Data Pasien           |")
11         print("| 5. Keluar Program             |")
12     else:
13         print("| 1. Tambah Data Pasien          |")
14         print("| 2. Lihat Data Pasien           |")
15         print("| 3. Hapus Data Pasien           |")
16         print("| 4. Keluar Program             |")
17     print("=====")
18
19     pilihan = input("pilih menu: ")
```

Program untuk menu pilihan

```

1  if pilihan == "1":
2      nama = input("nama Pasien  : ")
3      tinggi = int(input("tinggi Badan  : "))
4      bb = int(input("berat Badan  : "))
5      print("masukkan penyakit satu per satu ketik 'selesai' jika sudah")
6      penyakit_list = []
7      while True:
8          penyakit = input("Penyakit: ")
9          if penyakit.lower() == "selesai":
10             break
11             penyakit_list.append(penyakit)
12
13     beratIdeal, status = hitung_status(tinggi, bb)
14     pasien_dict[nama] = {
15         "tinggi": tinggi,
16         "berat": bb,
17         "ideal": beratIdeal,
18         "status": status,
19         "penyakit": penyakit_list
20     }

```

Program untuk pilihan 1



```
1 elif pilihan == "2":
2     print("\n-----")
3     print("|          DATA PASIEN          |")
4     print("-----")
5
6     if not pasien_dict:
7         print("belum ada pasien")
8     else:
9         nomor = 1
10        for nama in pasien_dict:
11            data = pasien_dict[nama]
12            print(f"\n[{nomor}] {nama}")
13            print("Tinggi Badan :", data["tinggi"])
14            print("Berat Badan  :", data["berat"])
15            print("Berat Ideal  :", data["ideal"])
16            print("Status       :", data["status"])
17            print("Penyakit      :")
18            for py in data["penyakit"]:
19                print(" -", py)
20            print("-----")
21            nomor = nomor + 1
```


Program untuk pilihan 2

```

1 elif pilihan == "3" and is_dokter:
2     if not pasien_dict:
3         print("belum ada data untuk diubah")
4     else:
5         print("DAFTAR PASIEN:")
6         nama_list = list(pasien_dict.keys())
7         nomor = 1
8         for nama in nama_list:
9             print(str(nomor) + ". " + nama)
10            nomor = nomor + 1
11        ubah = int(input("masukkan nomor pasien yang ingin diubah: ")) - 1
12        if 0 <= ubah < len(nama_list):
13            nama_pilih = nama_list[ubah]
14            data = pasien_dict[nama_pilih]
15            print("masukkan data baru (kosongkan jika tidak ingin diubah):")
16            nama_baru = input("Nama baru : ")
17            if nama_baru == "":
18                nama_baru = nama_pilih
19            tinggi_baru = input("Tinggi baru : ")
20            if tinggi_baru == "":
21                tinggi_baru = data["tinggi"]
22            else:
23                tinggi_baru = int(tinggi_baru)
24            berat_baru = input("Berat baru : ")
25            if berat_baru == "":
26                berat_baru = data["berat"]
27            else:
28                berat_baru = int(berat_baru)
29            print("Masukkan penyakit baru satu per satu (ketik 'selesai' jika tidak ingin menambah:")
30            penyakit_baru = []
31            while True:
32                p = input("Penyakit: ")
33                if p.lower() == "selesai":
34                    break
35                penyakit_baru.append(p)
36            if not penyakit_baru:
37                penyakit_baru = data["penyakit"]
38            beratIdeal, status = hitung_status(tinggi_baru, berat_baru)
39            del pasien_dict[nama_pilih]
40            pasien_dict[nama_baru] = {
41                "tinggi": tinggi_baru,
42                "berat": berat_baru,
43                "ideal": beratIdeal,
44                "status": status,
45                "penyakit": penyakit_baru
46            }
47            print("data pasien berhasil diperbarui")
48        else:
49            print("nomor tidak valid")

```

Program untuk pilihan 3




```

1 elif (pilihan == "4" and is_dokter) or (pilihan == "3" and not is_dokter):
2     if not pasien_dict:
3         print("tidak ada data pasien")
4     else:
5         print("DAFTAR PASIEN:")
6         nama_list = list(pasien_dict.keys())
7         nomor = 1
8         for nama in nama_list:
9             print(str(nomor) + ". " + nama)
10            nomor = nomor + 1
11        hapus = int(input("masukkan nomor pasien yang ingin dihapus: ")) - 1
12        if 0 <= hapus < len(nama_list):
13            nama_hapus = nama_list[hapus]
14            del pasien_dict[nama_hapus]
15            print(f"data pasien '{nama_hapus}' berhasil dihapus")
16        else:
17            print("nomor tidak valid.")

```

Program untuk pilihan 4



```

1 elif (pilihan == "5" and is_dokter) or (pilihan == "4" and not is_dokter):
2     print("pendataan pasien selesai")
3     break

```

Program untuk pilihan 5

4. Hasil Output

```
=== SILAHKAN LOGIN ===
Username: fakih
Password: 009
login berhasil

=====
|          PENDATAAN PASIEN (CRUD)          |
=====
| 1. Tambah Data Pasien                     |
| 2. Lihat Data Pasien                     |
| 3. Ubah Data Pasien                      |
| 4. Hapus Data Pasien                    |
| 5. Keluar Program                       |
=====
pilih menu: 1
nama Pasien   : nama
tinggi Badan  : 170
berat Badan   : 60
masukkan penyakit satu per satu ketik 'selesai' jika sudah
Penyakit: batuk, pilek, pusing
```

Hasil dari pilihan 1

```
pilih menu: 2

-----
|          DATA PASIEN          |
-----

[1] nama
Tinggi Badan : 170
Berat Badan  : 60
Berat Ideal   : 70
Status       : Berat badan ideal
Penyakit      :
- batuk, pilek, pusing
-----
```

Hasil dari pilihan 2

```

pilih menu: 3
DAFTAR PASIEN:
1. nama
masukkan nomor pasien yang ingin diubah: 1
masukkan data baru (kosongkan jika tidak ingin diubah):
Nama baru   : wahyu
Tinggi baru :
Berat baru  :
Masukkan penyakit baru satu per satu (ketik 'selesai' jika tidak ingin menambah):
Penyakit: selesai
data pasien berhasil diperbarui

```

Hasil dari pilihan 3

```

pilih menu: 4
DAFTAR PASIEN:
1. wahyu
masukkan nomor pasien yang ingin dihapus: 1
data pasien 'wahyu' berhasil dihapus

```

Hasil dari pilihan 4

```

=====
|          PENDATAAN PASIEN (CRUD)          |
=====
| 1. Tambah Data Pasien |
| 2. Lihat Data Pasien  |
| 3. Ubah Data Pasien   |
| 4. Hapus Data Pasien  |
| 5. Keluar Program     |
=====
pilih menu: 5
pendataan pasien selesai

```

Hasil dari pilihan 5

5. Langkah-langkah GIT

5.1 GIT Init

```

PS C:\Users\Hasmiyah\.vscode\praktikum-apd\post-test\post-test-apd-2> git init
Initialized empty Git repository in C:/Users/Hasmiyah/.vscode/praktikum-apd/post-test/post-test-apd-2/.git/
PS C:\Users\Hasmiyah\.vscode\praktikum-apd\post-test\post-test-apd-2>

```

Digunakan untuk menginisiasi repository Git ke lokal.

5.2 GIT Add

```
C:\Users\Hasmiyah\.vscode\praktikum-apd\post-test>git add .
```

Digunakan untuk menambahkan perubahan file ke staging area.

5.3 GIT Commit

```
C:\Users\Hasmiyah\.vscode\praktikum-apd\post-test>git commit -m "posttest 6"
[main 880f3b9] posttest 6
4 files changed, 171 insertions(+), 38 deletions(-)
delete mode 100644 post-test/post-test-apd-2/A1'25/pertemuan-3/pertemuan_3.py
create mode 100644 post-test/posttest_apd_5/tempCodeRunnerFile.py
create mode 100644 post-test/posttest_apd_6/2509106009_Muhammad_Fakih_Nabal_PT_6.py
```

Digunakan untuk menyimpan perubahan file yang sudah dimasukkan ke staging area.

5.4 GIT Remote

```
C:\Users\Hasmiyah\.vscode\praktikum-apd\post-test>git remote add origin https://github.com/fakih004/praktikum-apd.git
error: remote origin already exists.
```

Digunakan untuk menghubungkan repository lokal ke repository lain.

5.5 GIT Push

```
C:\Users\Hasmiyah\.vscode\praktikum-apd\post-test>git push origin main
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 2.15 KiB | 732.00 KiB/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/fakih004/praktikum-apd.git
 2f27b33..880f3b9  main -> main
```

Digunakan untuk mengirim commit repository lokal ke repository remote.