

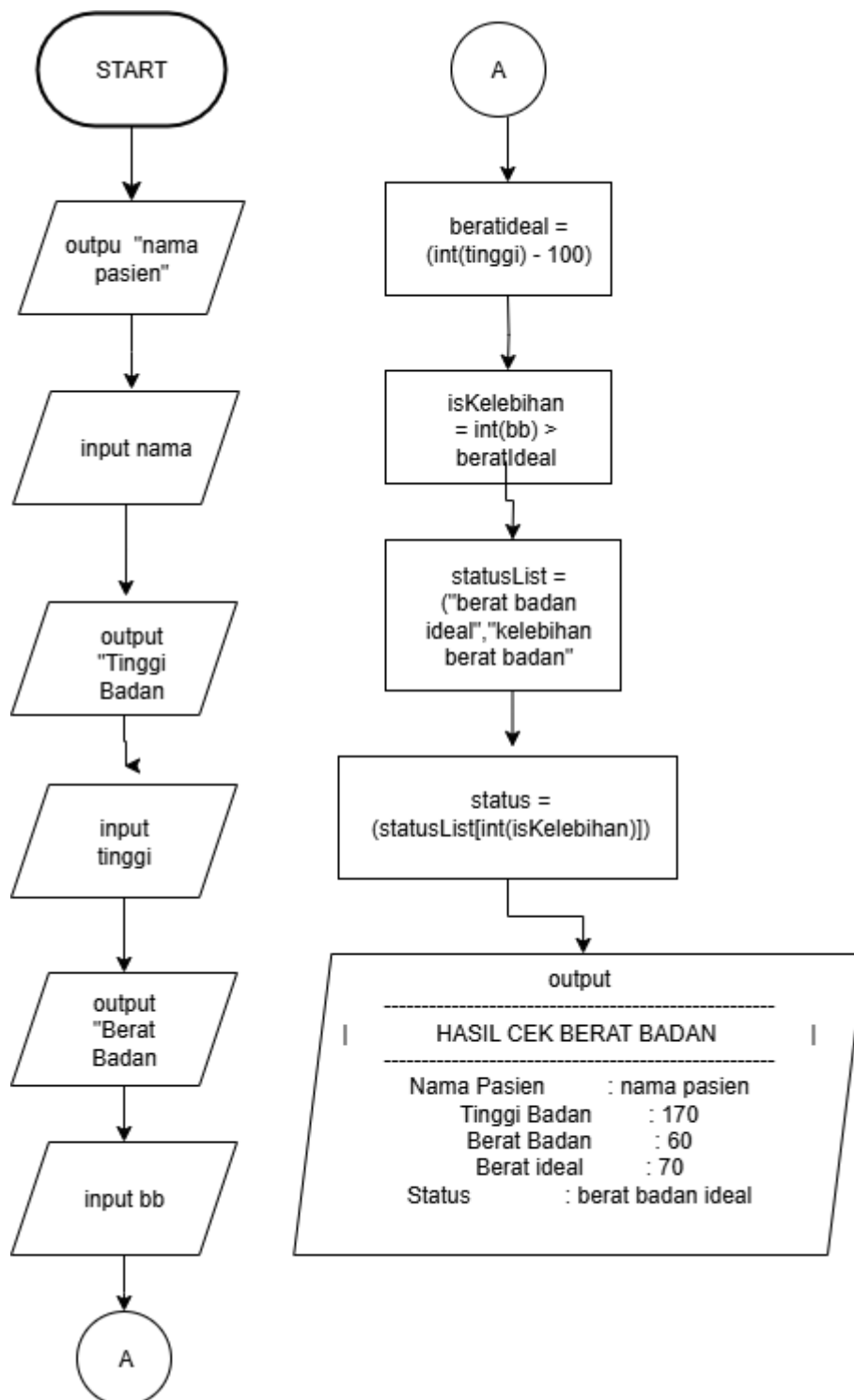
LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Muhammad Fakhri Nabal
2509106009
Kelas A1'25

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



Pertama - tama program akan menanyakan nama pasien yang akan diperiksa, nama yang dimasukkan oleh pasien akan ditampung variabel 'nama', program juga akan menanyakan tinggi badan pasien yang akan ditampung variabel 'tinggi', program juga akan menanyakan berat badan pasien yang akan ditampung variabel 'bb'.

Program akan mengolah data yang diberikan oleh pasien untuk menentukan berat ideal pasien, jika berat ideal sudah ditemukan ,maka program akan membandingkan berat badan pasien dengan berat ideal, jika berat badan pasien lebih besar dari berat ideal maka, program akan meng-output “kelebihan berat badan”, jika berat badan pasien kurang dari berat badan ideal maka program akan meng-output “berat badan ideal”, setelah menentukan pasien memiliki berat yang ideal atau tidak, program akan meng-output hasil pengolahan data.

2. Deskripsi Singkat Program

Program ini bertujuan untuk membantu pasien mengecek status berat badan berdasarkan tinggi badan dan berat badan, sehingga pasien mengetahui bahwa ia memiliki berat badan yang ideal atau tidak.

3. Source Code

```
Judul:
    Pengecekan berat badan
Deklarasi:
    nama,tinggi,bb,beratIdeal,isKelebihan,statusList,status
Algoritma:
1.      START
2.      Print "nama pasien "
3.      Input nama
4.      Print "Tinggi Badan"
5.      Input tinggi
6.      Print "Berat Badan"
7.      Input bb
8.      beratIdeal = (int(tinggi) - 100)
9.      isKelebihan = int(bb) > beratIdeal
10.     statusList = ("berat badan ideal",kelebihan berat badan")
11.     status = (statusList[int(isKelebihan)])
12.     print
13.     (""-----
14.     |          HASIL CEK BERAT BADAN          |
15.     -----)
16.     print("Nama Pasien          : {nama}")
17.     print("Tinggi Badan          : {tinggi}")
18.     print("Berat Badan           : {bb}")
19.     print('Berat Ideal           : '+ str(beratIdeal))
20.     print(f"Status               : {status}")
```

4. Hasil Output

```
Nama Pasien      : nama pasien
Tinggi Badan     : 170
Berat Badan      : 60
-----
|                HASIL CEK BERAT BADAN                |
-----
Nama Pasien      : nama pasien
Tinggi Badan     : 170
Berat Badan      : 60
Berat Ideal      : 70
Status           : berat badan ideal
```

5. Langkah-langkah GIT

5.1 GIT Init

Digunakan untuk menginisiasi repository Git baru ke lokal.

```
PS C:\Users\Hasmiyah\.vscode\praktikum-apd> git init
```

Initialized empty Git repository in C:/Users/Hasmiyah/.vscode/praktikum-apd/.git/

5.2 GIT Add

Digunakan untuk menambahkan berubahan file ke staging area.

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

5.3 GIT Commit

Digunakan untuk menyimpan perubahan file yang sudah di masukkan ke staging area.

PS C:\Users\Hasmiyah\.vscode\praktikum-apd> git commit -m "ada yang salah tadi bang ini ngirim lagi"

5.4 GIT Remote

Digunakan untuk menghubungkan repository lokal ke repository lain.

PS C:\Users\Hasmiyah\.vscode\praktikum-apd> git remote add origin https://github.com/fakih004/praktikum-apd.git

5.5 GIT Push

Digunakan untuk mengirim commit repository lokal ke repository remote.

PS C:\Users\Hasmiyah\.vscode\praktikum-apd> git push origin main

Enumerating objects: 7, done.

Counting objects: 100% (7/7), done.

Delta compression using up to 4 threads

Compressing objects: 100% (2/2), done.

Writing objects: 100% (4/4), 343 bytes | 49.00 KiB/s, done.

Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.
e.

Writing objects: 100% (4/4), 339 bytes | 339.00 KiB/s, done.

Total 4 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/fakih004/praktikum-apd.git

b2b86ee..3cff8a0 main -> main