

LAPORAN PRAKTIKUM POSTTEST 4
ALGORITMA DAN PEMROGRAMAN DASAR



Ananda Daffa Harahap B1

2409106050

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA

2024

LATAR BELAKANG

Studi kasus :

Pada posttest kali ini, kalian diminta untuk mengimplementasikan konsep perulangan pada program posttest kalian sebelumnya.

ketentuan :

1. - Tambahkan autentikasi pada program kalian berisi input username dan password sebelum masuk ke program menu utama kalian.
 - Username merupakan nama panggilan kalian (ex : ifnu) dan passwordnya berisi 3 digit nim terakhir kalian (ex : 060).
 - Jika salah menginputkan username/password 3x maka program akan langsung berhenti.
2. - Tambahkan perulangan pada menu program kalian yaitu posttest sebelumnya.
 - Program tidak akan berhenti sampai memilih menu keluar dari program.

Point +:

- Membersihkan layar terminal ketika program dijalankan menggunakan Built in Library Python yaitu os dan menggabungkannya dengan perintah terminal yaitu cls untuk Windows clear untuk Linux/Macos.
- Autentikasi tidak boleh input kosong atau spasi.
- Program tidak berhenti sebelum diminta atau tidak eror.

Dari studi kasus diatas, dapat kita ambil kesimpulan bahwa kita diminta untuk membuat dan memasukkan perulangan dan autentikasi pada program menghitung bangun ruang (NIM genap) yang kita buat kemarin dengan cara

membuat flowchart dan python.

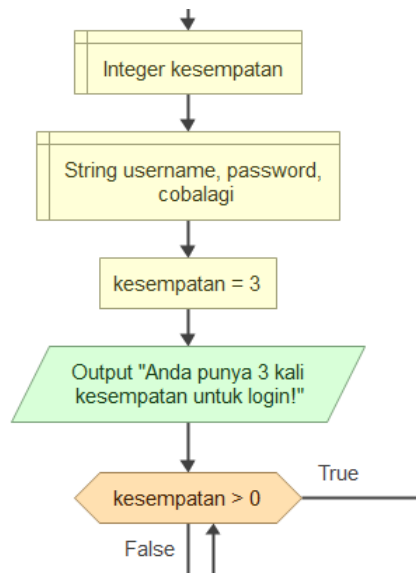
Pertama-tama, kita disuruh membuat dan memasukkan fitur autentikasi kedalam program menghitung bangun datar atau ruang sesuai dengan NIM kita (Genap) yang telah kita buat. Lalu kita memiliki peraturan pertama, dimana username dan password berupa nama panggilan kita (ex : daffa) dan passwordnya berupa 3 digit terakhir dari NIM kami (ex : 050). Peraturan kedua, adalah jika user salah menginputkan username atau password sebanyak 3 kali, maka program akan langsung berhenti. Terakhir peraturan ketiga, program tidak akan berhenti sampai user memilih menu keluar dari program.

SOLUSI

Solusi yang saya terapkan adalah dengan cara pertama-tama membuat flowchart tentang studi kasus diatas lalu dilanjutkan dengan membuat program studi kasus diatas dengan bahasa pemrograman python yang dimana saya membuat autentikasi terlebih dahulu lalu dimasukkan ke dalam program 7 bangun ruang yang kita buat kemarin.

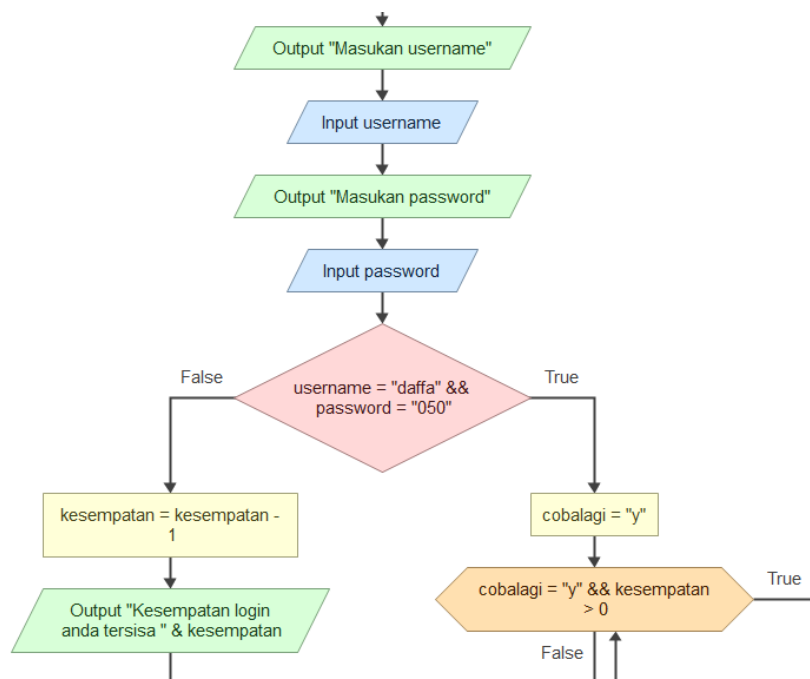
A. Flowchart

Sebelum membuat suatu program kita harus membuat flowchartnya terlebih dahulu, fungsinya adalah untuk mempermudah kita menentukan atau memahami alur program yang akan kita buat.



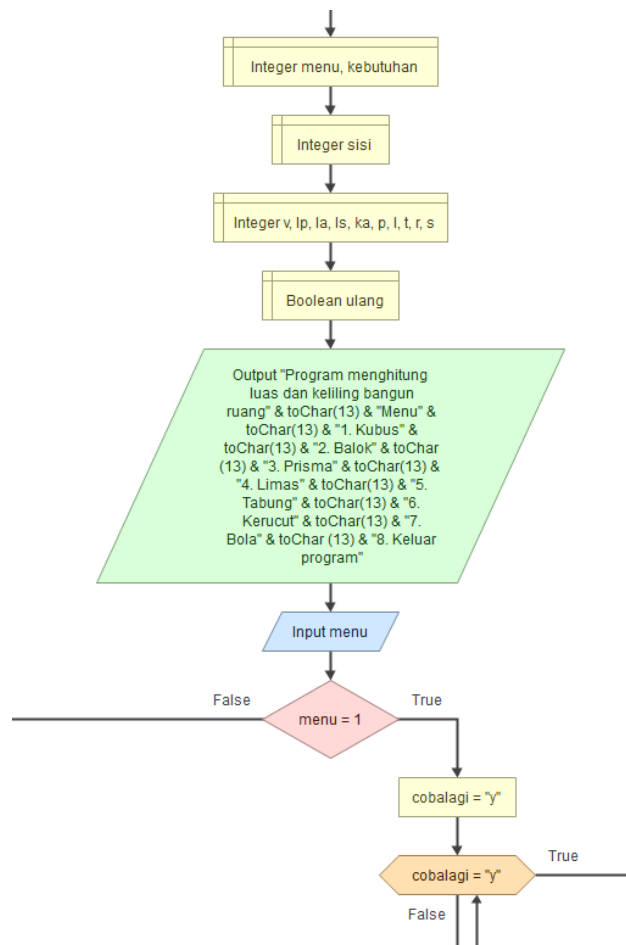
Flowchart Gambar 1

Dilihat dari **Flowchart Gambar 1**, 4 langkah pertama yang saya lakukan adalah deklarasi tipe data, variabel, serta assign variabel kesempatan sebanyak 3 agar saat masuk ke perulangan while, akan bernilai true karena kondisi perulangan while yang berisi “kesempatan > 0” sementara kesempatan bernilai 3 yang mana akan menghasilkan output “True” karena 3 lebih besar dari 0, apabila nilai kesempatan = 0 maka akan menghasilkan output “False”.



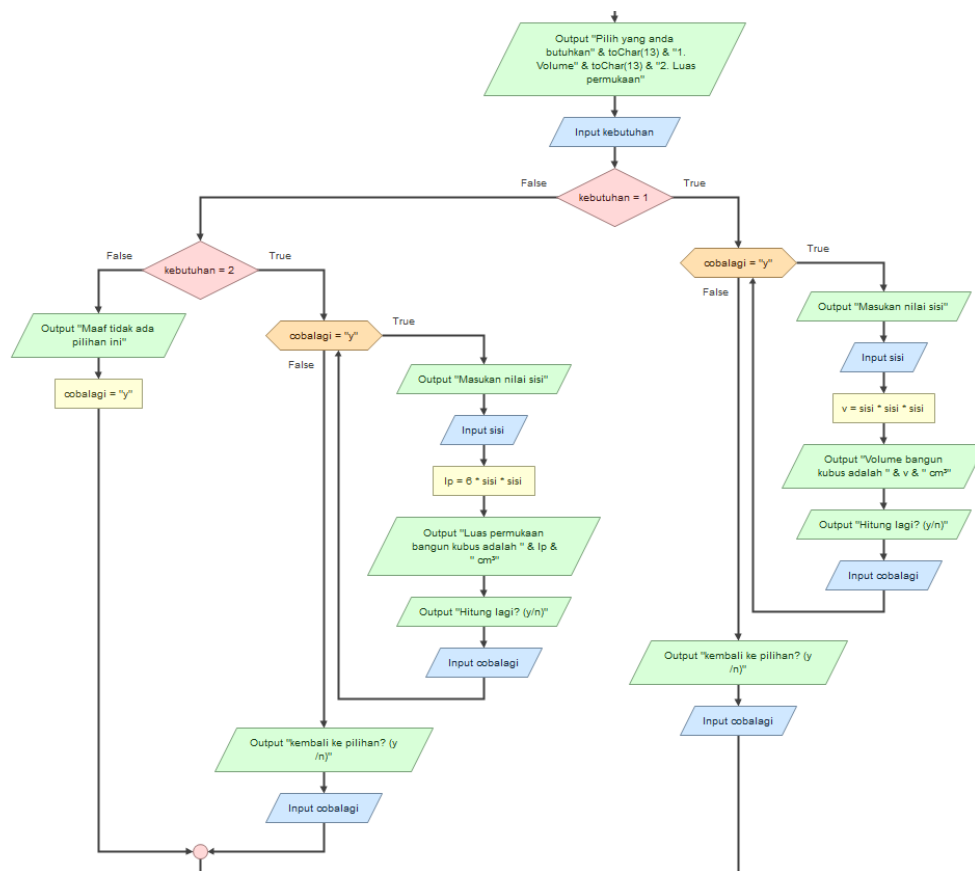
Flowchart Gambar 2

Pada **Flowchart Gambar 2**, disinilah terjadinya proses autentikasi, apabila input username dan password yang input benar maka nilai output akan bernilai “True” yang akan masuk kedalam program menghitung bangun ruang yang telah kita buat kemarin, apabila salah input username dan password maka akan menghasilkan output “False” yang mana akan terjadinya proses dimana variabel kesempatan yang jumlah awalnya adalah 3 akan dikurangi 1 sesuai dengan peraturan soal yang berbunyi “jika user salah menginputkan username atau password sebanyak 3 kali, maka program akan langsung berhenti”. Lalu akan ditampilkan output berupa pesan yang berbunyi “Kesempatan login anda tersisa (jumlah yang tersisa karena nilai variabel kesempatan dikurangi 1 setiap salah input)” dan akan disuruh menginput ulang hingga kesempatan yang dimiliki oleh user itu habis (berjumlah 0).



Flowchart Gambar 3

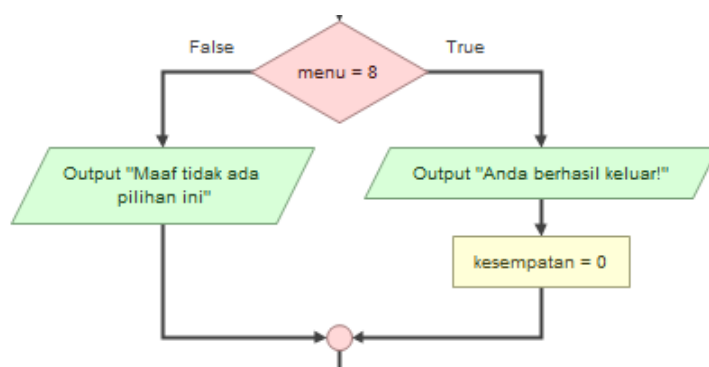
Pada **Flowchart Gambar 3**, terjadi hal yang sama seperti **Flowchart Gambar 1**, yang mana terjadi deklarasi serta process yang sama seperti post test sebelumnya. Tetapi perbedaan disini dengan adanya perulangan “While”, yang dimana hal pertama yang akan dilakukan setelah memilih menu (1-7) adalah “assign” atau membuat variabel “cobalagi” bernilai “y” atau “True” yang akan masuk kedalam program perhitungan bangun ruang. Apabila variabel “cobalagi” bernilai selain “y” maka ia akan menghasilkan output “False” yang akan menjalankan proses berikutnya.



Flowchart Gambar 4

Pada **Flowchart Gambar 4**, dapat dilihat adanya perulangan “While” yang dimana akan terjadi setelah user selesai melakukan perhitungan baik itu volume maupun luas permukaan, dengan langkah pertama akan menanyakan kepada user

apakah user ingin melakukan perhitungan ulang, apabila user menjawab “y” maka program akan menjalankan perhitungan lagi sesuai dengan kebutuhan user. Tetapi apabila user menjawab “n”, maka program mengubah variabel “cobalagi” yang awalnya bernilai “y” menjadi “n” sesuai dengan input user, setelah dicek di perulangan “While”, apabila variabel “cobalagi” tidak bernilai “y”. Maka akan menghasilkan output “False” yang akan memberikan output “Apakah anda ingin kembali ke pilihan?” yang dimaksud dengan pilihan disini adalah pilihan 1. Volume dan 2. Luas permukaan yang ada di output pertama. Begitu juga untuk “kebutuhan = 2” atau pilihan 2. Luas permukaan, tetapi disini saya membuat agar apabila user menginput selain yang ada dipilihan (1. Volume 2. Luas permukaan) maka akan memunculkan output “Maaf tidak ada pilihan ini” dan akan meng “assign” nilai variabel “cobalagi” menjadi “y” yang akan membuat perulangan dimana user akan dikembalikan ke menu pilihan (1. Volume 2. Luas permukaan). Hal ini juga berlaku untuk seluruh variabel “Menu” lainnya serta variabel “kebutuhan”



Flowchart Gambar 5

Untuk **Flowchart Gambar 5**, variabel “menu = 8” adalah menu untuk keluar apabila user menginput “8” pada saat memilih menu maka akan otomatis menghasilkan nilai “True” yang akan menampilkan output “Anda berhasil keluar!”, lalu akan terjadi proses “assign” dimana akan membuat variabel “kesempatan” menjadi bernilai 0. Yang dimana ini akan membuat program berhenti karena untuk masuk kedalam menu terdapat 2 syarat yaitu, variabel “cobalagi” bernilai “y” dan variabel “kesempatan” lebih besar dari 0.

B. Code

A screenshot of a code editor with a dark background and light-colored text. The code is written in Python and is numbered from 1 to 22. It includes comments in Indonesian and uses standard Python syntax for loops, conditionals, and input/output. The code is as follows:

```
1 # Autentikasi
2 kesempatan = 3
3 print("="*width)
4 print("Anda punya 3 kali kesempatan untuk login!".center(width))
5 while kesempatan > 0:
6     print("="*width)
7
8     # User diminta memasukkan username dan password
9     username = input("Masukan username : ").strip()
10    password = input("Masukan password : ").strip()
11
12    if not username or not password:
13        print("="*width)
14        print("Username dan Password tidak boleh kosong atau spasi!")
15        continue
16
17    if username == "daffa" and password == "050":
18
19        # Ini agar saat user ingin keluar dia tidak kembali ke menu sebelumnya
20        cobalagi = "y"
21
22        while cobalagi == "y" and kesempatan > 0:
```

Kode 1

Berdasarkan gambar **Kode 1**, yang pertama kita “assign” atau menyatakan bahwa variabel “kesempatan” akan memiliki nilai 3, setelah dicek oleh perulangan “while” dimana variabel “kesempatan” lebih besar dari 0, maka user akan dapat melakukan login untuk masuk kedalam program menghitung bangun ruang.

Pada blok kode username dan password terdapat kode yang berbunyi “.strip()”, fungsi dari block kode ini adalah agar user tidak dapat menginput spasi atau kosong didalam username dan password.

Terdapat blok kode “if not username or not password:”, fungsi dari blok kode ini adalah agar saat user menginput spasi atau kosong, maka akan dicek dulu oleh blok kode ini, apabila ternyata tidak sama dengan yang diminta pada blok kode username dan password maka akan diminta untuk menginput kembali tanpa spasi atau tidak boleh kosong begitu saja.

Blok kode “continue” berfungsi untuk membuat user lanjut ke blok kode berikutnya apabila user telah menginput sesuai dengan ketentuan yang ada.

Terdapat variabel “cobalagi” yang bernilai “y” serta blok kode perulangan

“while” yang dimana ini adalah awal dari perulangan didalam program ini.

```
1  os.system("cls")
2
3      print("="*width)
4      # Program untuk menghitung bangun ruang
5      print("Menu Program Menghitung Bangun Ruang".center(width))
6
7      print("=" * width)
8
9      print("1. Kubus \n2. Balok \n3. Prisma \n4. Limas \n5. Tabung \n6. Kerucut \n7. Bola \n8. Keluar program")
10
11     print("="*width)
12
13     # Masukan Input Pengguna
14     menu = int(input("Pilih Menu : "))
15
16     # Ini untuk menjalankan perintah clear screen untuk terminal
17     os.system("cls")
18
```

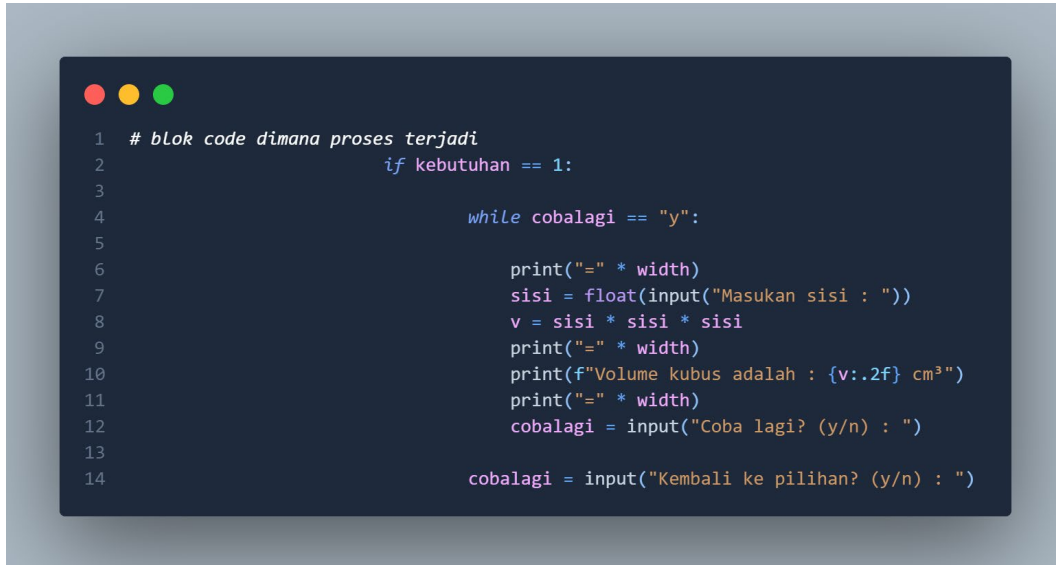
Kode 2

Pada gambar **Kode 2**, akan muncul menu sesuai program yang kemaren kita buat.

```
1  # Memulai perulangan
2      cobalagi = "y"
3
4      while cobalagi == "y":
5
6          # Menu pertama yaitu untuk menghitung kubus
7          if menu == 1: # Menu kubus
8              print("=" * width)
9              print("Pilih kebutuhan".center(width))
10             print("=" * width)
11             print("1. Volume \n2. Luas permukaan")
12             print("=" * width)
13             kebutuhan = int(input("Pilih : "))
14
15             # Ini untuk menjalankan perintah clear screen untuk terminal
16             os.system("cls")
17
```

Kode 3

Pada gambar **Kode 3**, dapat dilihat dimulainya perulangan “while”.

A screenshot of a code editor with a dark background and light-colored text. The code is written in Python and is numbered from 1 to 14 on the left side. The code starts with a comment on line 1: "# blok code dimana proses terjadi". On line 2, there is an if statement: "if kebutuhan == 1:". Inside this if block, on line 4, there is a while loop: "while cobalagi == 'y':". The while loop contains several lines of code: line 6: "print('=' * width)", line 7: "sisi = float(input('Masukan sisi : '))", line 8: "v = sisi * sisi * sisi", line 9: "print('=' * width)", line 10: "print(f'Volume kubus adalah : {v:.2f} cm³')", line 11: "print('=' * width)", line 12: "cobalagi = input('Coba lagi? (y/n) : ')", and line 14: "cobalagi = input('Kembali ke pilihan? (y/n) : ')".

```
1  # blok code dimana proses terjadi
2      if kebutuhan == 1:
3
4          while cobalagi == "y":
5
6              print("=" * width)
7              sisi = float(input("Masukan sisi : "))
8              v = sisi * sisi * sisi
9              print("=" * width)
10             print(f"Volume kubus adalah : {v:.2f} cm³")
11             print("=" * width)
12             cobalagi = input("Coba lagi? (y/n) : ")
13
14             cobalagi = input("Kembali ke pilihan? (y/n) : ")
```

Kode 4

Digambar **Kode 4**, ini adalah blok kode yang dimana program perhitungan bangun ruang kita akan berjalan.

Serta terdapat blok kode variabel “cobalagi” dimana yang pertama bertanya apakah user ingin menghitung lagi, dan yang kedua adalah apabila user ingin kembali ke pilihan (1. Volume 2. Luas permukaan). Untuk sisa program perhitungan bangun ruangnya memiliki struktur dan cara yang sama yang membedakan hanya rumusnya saja.

```

1  elif menu == 8:
2      kesempatan = 0
3      print("=" * width)
4      print("Anda berhasil keluar!".center(width))
5      print("=" * width)
6
7      sleep(2)
8
9      os.system("cls")
10     break

```

Kode 5

Pada **Kode 5**, dapat dilihat bahwa yang akan terjadi pertama kali adalah membuat variabel “kesempatan” menjadi bernilai “0”, yang dimana apabila variabel “kesempatan” bernilai lebih kecil atau sama dengan “0” maka user akan otomatis selesai programnya karena variabel “menu == 8” adalah untuk keluar dari program. Karena itu terdapat blok kode *break* yang dimana kode ini berfungsi untuk menghentikan program langsung setelah kode ini dijalankan.

```

1  else:
2
3      os.system("cls")
4
5      kesempatan -= 1
6      print("=" * width)
7      print(f"Kesempatan login anda sisa {kesempatan}".center(width))

```

Kode 6

Pada **Kode 6**, ini adalah akhir kode program ini dimana blok kode ini bertujuan untuk apabila user salah menginput username atau password maka akan dikurangi kesempatannya sebanyak 1, sehingga saat awal ia memiliki kesempatan sebanyak 3 akan berkurang apabila dia salah memasukkan username dan password.