

LAPORAN PRAKTIKUM

PEMROGRAMAN KOMPUTER (PYTHON)



Disusun Oleh:
Rafli Pratama
H1101241008

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS TANJUNGPURA PONTIANAK
2024

Praktikum 4

Repetition

1.1. Dasar Teori

1.1.1 Instruksi Kontrol Perulangan

Pengulangan(looping) adalah suatu bagian/blok yang bertugas melakukan kegiatan mengulang suatu proses sesuai dengan yang diinginkan. looping merupakan proses berulang dari sekumpulan intruksi dalam kode program sampai kondisi yang ditentukan terpenuhi. Looping memungkinkan program untuk melakukan tugas yang sama berulang kali tanpa harus menuliskan kode yang sama. Banyak dari aplikasi perangkat lunak yang melakukan pekerjaan berulang sampai sebuah kondisi yang diinginkan, oleh karena itu pengulangan merupakan bagian yang penting dalam pemrograman karena dengan adanya pengulangan pembuat program tidak perlu menulis kode program sebanyak pengulangan yang diinginkan.

Pengulangan mempunyai beberapa bagian yang harus dipenuhi yaitu:

1. Inisialisasi adalah tahap persiapan membuat kondisi awal sel melakukan pengulangan, misalnya mengisi variabel dengan nilai awal. Tahap ini dilakukan sebelum memasuki bagian pengulangan.
2. Proses terjadi di dalam bagian pengulangan dimana berisi semua proses yang perlu dilakukan secara berulang-ulang.
3. Iterasi terjadi di dalam pengulangan di mana merupakan kondisi pertambahan agar pengulangan dapat terus berjalan.
4. Terminasi adalah kondisi berhenti dari pengulangan, kondisi berhenti sangat penting dalam pengulangan agar pengulangan dapat berhenti, tidak menjadi pengulangan yang tanpa henti. Kondisi pengulangan adalah kondisi yang dipenuhi oleh kondisi jalannya algoritma untuk masuk ke dalam blok pengulangan.

Pengulangan merupakan salah satu inti dari analisis kasus pada pembuatan algoritma, sebuah kasus harus dipikirkan penyelesaiannya dengan pemikiran ada proses atau aksi yang harus dikerjakan secara berulang agar sebuah kasus terselesaikan. Pernyataan pengulangan disebut loop, dan digunakan untuk mengulangi kode yang sama beberapa kali berturut-turut.

Python memiliki 2 jenis looping:

1. Condition controlled loop atau perulangan yang dikontrol kondisi dalam Python adalah perulangan yang mengeksekusi blok kode secara berulang selama kondisi tertentu terpenuhi. Kondisi ini akan diuji pada awal setiap iterasi, dan jika kondisi terpenuhi, iterasi akan berlanjut. Perulangan yang dikontrol kondisi digunakan ketika jumlah iterasi yang dibutuhkan tidak diketahui.

while loop dalam Python adalah jenis perulangan yang terus menjalankan blok kode di dalamnya selama kondisi tertentu bernilai True. Ini berarti loop akan terus berulang sampai kondisi tersebut menjadi False.

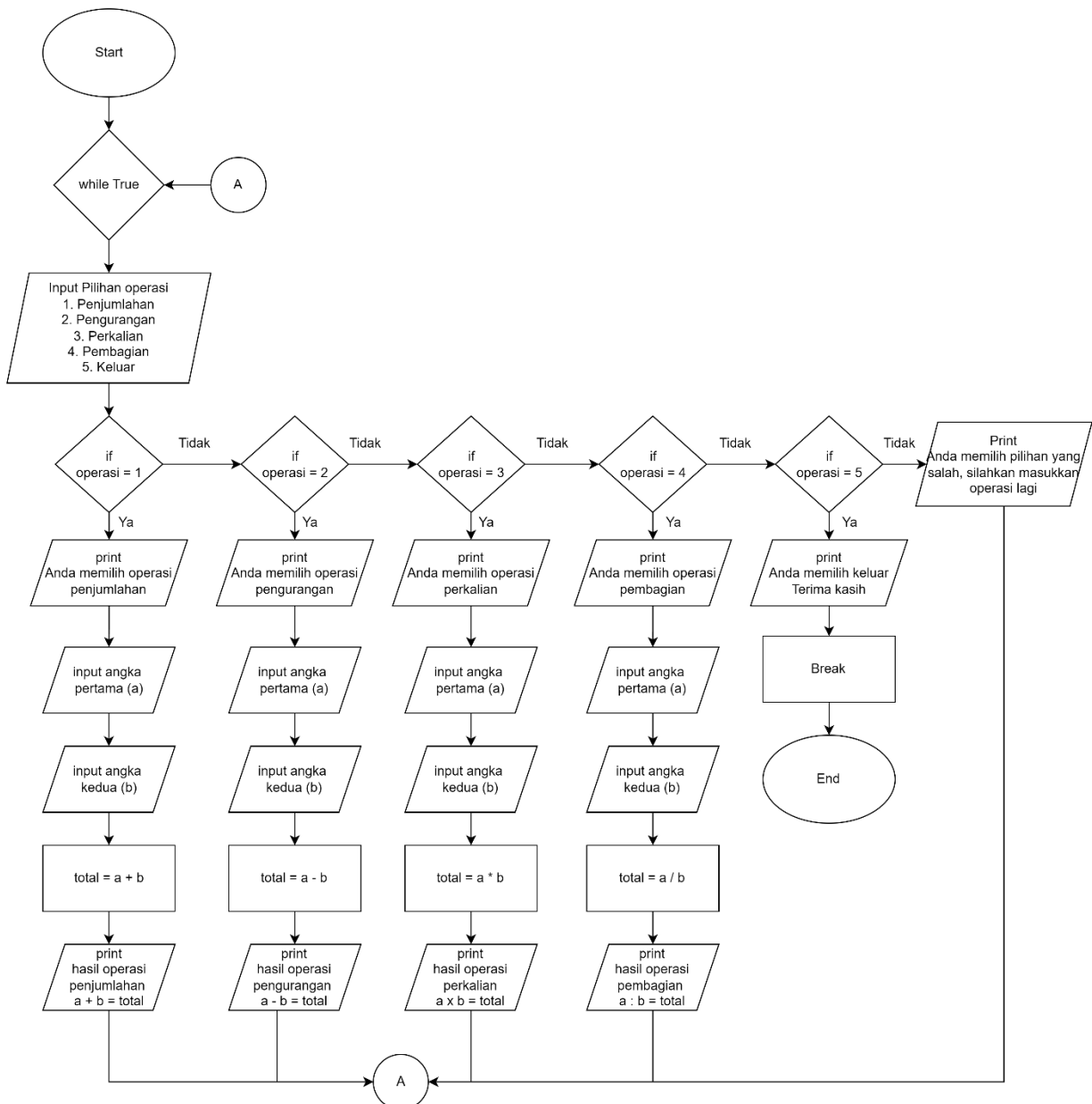
2. Count-controlled loop atau perulangan yang dikontrol hitungan dalam Python adalah konsep pemrograman yang digunakan untuk mengeksekusi blok kode sejumlah kali tertentu. Perulangan ini menggunakan variabel penghitung untuk melacak berapa kali algoritme telah melakukan iterasi.

Perulangan for dalam Python adalah konstruksi yang digunakan untuk mengiterasi atau mengulang suatu urutan (sequence) atau objek yang dapat diiterasi. for loop memungkinkan kita untuk menjalankan satu blok kode berulang kali untuk setiap elemen dalam urutan tersebut.

1.2. Percobaan Praktikum

1.2.1 Soal 1

Buatlah flowchart dan program untuk menghitung operasi aritmatika dengan pilihan fitur menu



```

1 while True:
2     operasi =int(input('Pilih Operasi Aritmatika : 1. Penjumlahan \n2. Pengurangan \n
3     3. Perkalian \n4. Pembagian \n5. Keluar'))
4     if operasi == 1:
5         print('Anda memilih operasi penjumlahan')
6         a = int(input('Masukan angka pertama : '))
7         b = int(input('Masukan angka kedua : '))
8         total = a+b
9         print(f'Hasil operasi penjumlahan {a} + {b} = {total}')
10        print('-----')
11    elif operasi == 2 :
12        print('Anda memilih operasi pengurangan')
13        a = int(input('Masukan angka pertama : '))
14        b = int(input('Masukan angka kedua : '))
15        total = a-b
16        print(f'Hasil operasi pengurangan {a} - {b} = {total}')
17        print('-----')
18    elif operasi == 3 :
19        print('Anda memilih operasi perkalian')
20        a = int(input('Masukan angka pertama : '))
21        b = int(input('Masukan angka kedua : '))
22        total = a*b
23        print(f'Hasil operasi perkalian {a} x {b} = {total}')
24        print('-----')
25    elif operasi == 4:
26        print('Anda memilih operasi perkalian')
27        a = int(input('Masukan angka pertama : '))
28        b = int(input('Masukan angka kedua : '))
29        total = a/b
30        print(f'Hasil operasi perkalian {a} : {b} = {total}')
31        print('-----')
32    elif operasi == 5:
33        print('Anda memilih keluar! \nTerima kasih')
34        print('-----')
35        break
36    else:
37        print('Anda memilih pilihan yang salah! \nSilahkan pilih operasi kembali')
38        print('-----')

```

Output :

```

Anda memilih pilihan yang salah!
Silahkan pilih operasi kembali
-----
Anda memilih operasi penjumlahan
Hasil operasi penjumlahan 12 + 3 = 15
-----
Anda memilih operasi pengurangan
Hasil operasi pengurangan 12 - 3 = 9
-----
Anda memilih operasi perkalian
Hasil operasi perkalian 12 x 3 = 36
-----
Anda memilih operasi perkalian
Hasil operasi perkalian 12 : 3 = 4.0
-----
Anda memilih keluar!
Terima kasih
-----

```

Penjelasan :

Pada program ini kita menggunakan while dengan kondisi True sebagai perulangan dan tidak akan berhenti sampai kita melakukan break, setelah memasuki perulangan while, user diminta untuk memasukkan inputan integer 1 – 5, yang disimpan pada variabel operasi, jika operasi = 1, maka akan masuk ke operasi penjumlahan, jika operasi = 2 akan masuk ke operasi pengurangan, jika operasi = 3 maka akan masuk ke dalam operasi perkalian, jika operasi = 4 maka akan masuk ke operasi pembagian, jika operasi = 5, maka akan menampilkan print bahwa user memilih keluar dan melakukan break yang akan menghentikan perulangan while, jika operasi tidak dari kondisi di atas maka user akan di minta memasukkan kembali pilihan operasi, pada operasi = 1-4, didalam masing masing kondisi user akan diminta memasukkan dua buah input bilangan yang dimana bilangan tersebut akan di operasikan sesuai dengan kondisi 1-4, setelah itu akan di tampilkan output hasil perhitungannya.

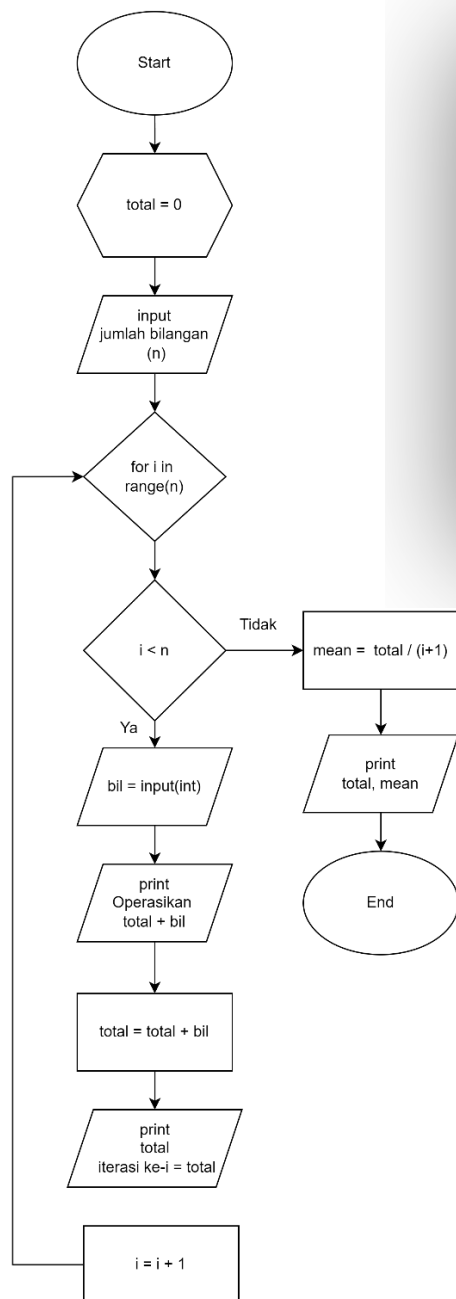
1.2.2 Soal 2

Buat flowchart dan program yang menerima input data sebanyak N, menghitung nilai total hasil penjumlahan, dan menghitung rata-rata nya

Input: N, data ke-1 s/d ke-N

Output: Total, RataRata

Proses: $\text{Total} = \sum \text{datai}$, $\text{RataRata} = \text{Total}/N$



```

1  n = int(input('Masukkan jumlah bilangan : '))
2  total = 0
3  for i in range(n):
4      bil = int(input('Masukkan bilangan : '))
5      print(f'Operasikan {total} + {bil}')
6      total += bil
7      print(f'Total pada iterasi ke-{i+1} adalah = {total}')
8      print('-----')
9  mean = int(total / (i+1))
10 print(f'Total keseluruhan adalah : {total} \n
    Dengan rata-ratanya adalah {mean}')
11
  
```

Output :

```

Operasikan 0 + 12
Total pada iterasi ke-1 adalah = 12
-----
Operasikan 12 + 12
Total pada iterasi ke-2 adalah = 24
-----
Operasikan 24 + 12
Total pada iterasi ke-3 adalah = 36
-----
Operasikan 36 + 12
Total pada iterasi ke-4 adalah = 48
-----
Operasikan 48 + 12
Total pada iterasi ke-5 adalah = 60
-----
Total keseluruhan adalah : 60
Dengan rata-ratanya adalah 12
  
```

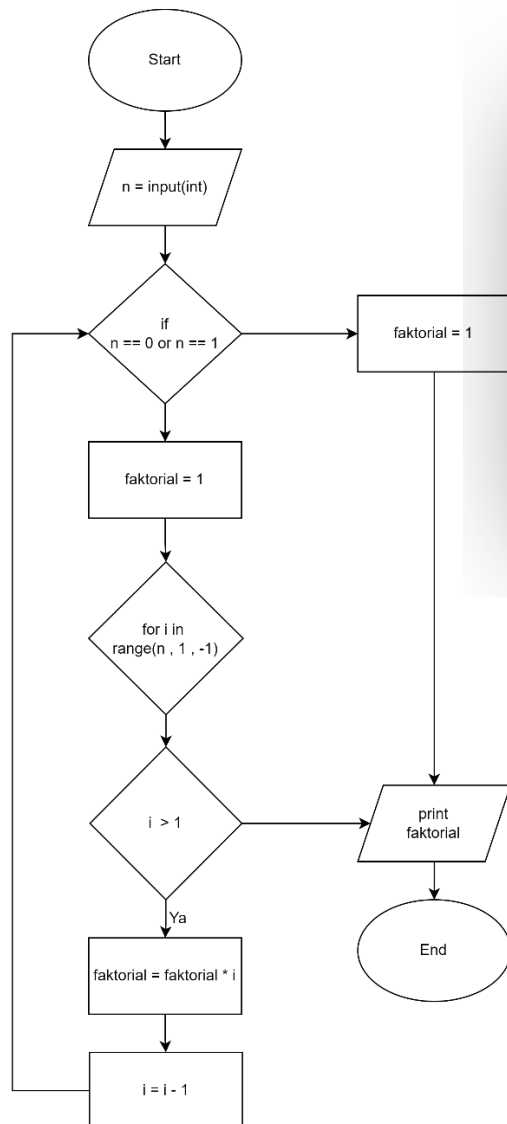
Penjelasan :

Pada program ini user di minta untuk memasukkan inputan yang merupakan batas index untuk perulangan for, setelah masuk ke dalam perulangan for dengan range (n), user akan di minta memasukkan inputan bilangan, setelah itu akan dilakukan proses perhitungan total + bilangan, lalu di tampilkan total pada saat iterasi ke-1, setelah perulangan mencapai index ke-n, perulangan for akan keluar dan dilakukan perhitungan mean atau rata rata dengan rumus total / (i+1), setelah itu di print total terakhir dan rata rata dari seluruh bilangan yang di inputkan.

1.2.3 Soal 3

Buatlah flowchart dan program untuk menghitung nilai faktorial Faktorial $(n) = n! = n \times (n-1)! = n \times (n-1) \times (n-2) \times \dots \times (n-(n-1))$

Misalnya: jika $n=4$, maka $4! = 4 \times 3 \times 2 \times 1$



```
1 n = int(input('Masukkan bilangan faktorial : '))
2 if n == 0 or n == 1:
3     faktorial = 1
4 else:
5     faktorial = 1
6     for i in range(n, 1, -1):
7         faktorial *= i
8
9 print(f'Faktorial dari {n} adalah {faktorial}')
```

Output :

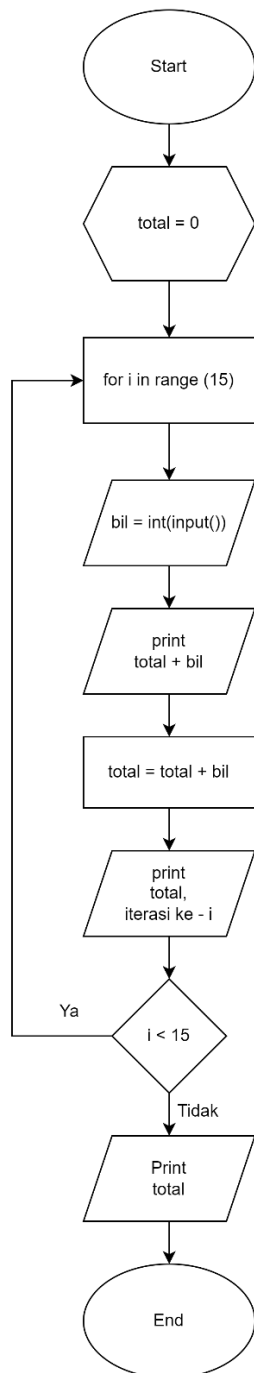
Faktorial dari 4 adalah 24

Penjelasan :

Pada program ini user akan di minta untuk memasukkan bilangan n sebagai bilangan yang akan di cari faktorialnya, lalu di lakukan pengkondisian apakah bilangan yang di input = 0 atau = 1, jika iya maka faktorial = 1 dan program akan selesai, namun jika tidak, maka akan dilaukan perulangan for sebanyak n atau for in in range (n ,1 , -1) artinya dari index n ke 1, dengan decrement sebanyak -1, didalam pengulangan tersebut, akan di lakukan proses perhitungan faktorial yakni, faktorial = faktorial x i, jika i sudah < 1, maka perulangan selesai dan menampilkan output faktorial

1.2.4 Soal 4

Buatlah flowchart dan program yang menerima masukan 15 buah bilangan integer dan mencetak jumlah total dari 15 bilangan tersebut!



```

1 total_bil = 0
2 for i in range (15):
3     bil = int(input('Masukkan bilangan : '))
4     print(f'Operasikan {total_bil} + {bil}')
5     total_bil += bil
6     print(f'Iterasi ke- {i+1}')
7     print(f'Total saat ini : {total_bil}')
8     print('-----')
9     print(f'Total seluruh bilangan adalah : {total_bil}')
  
```

Output :

```

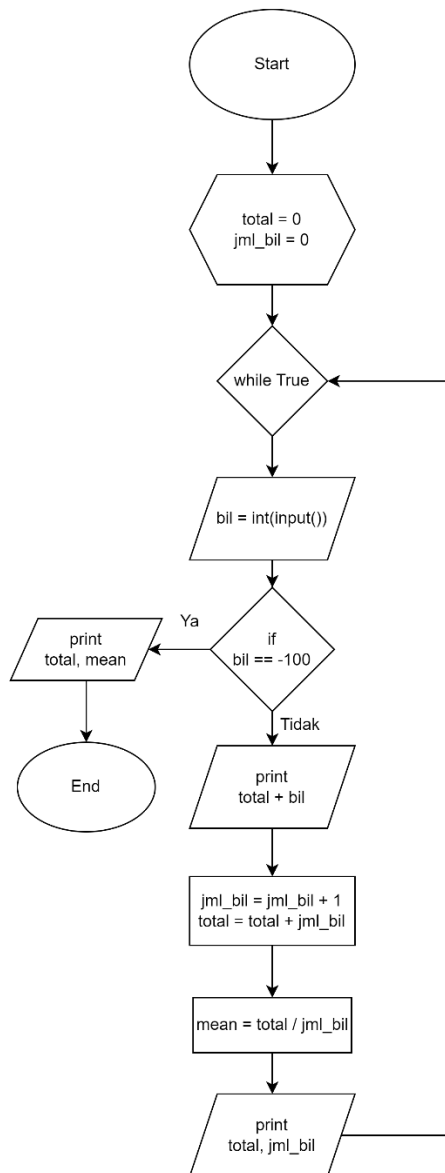
-----
Operasikan 113 + 12
Iterasi ke- 6
Total saat ini : 125
-----
Operasikan 125 + 23
...
Iterasi ke- 15
Total saat ini : 244
-----
Total seluruh bilangan adalah : 244
  
```

Penjelasan :

Dalam program ini akan dilakukan perulangan iterasi dengan menggunakan for, dengan index sebanyak 15, artinya setiap $i < 15$ bernilai True user akan diminta memasukkan inputan bertipe integer, lalu bilangan tersebut akan di tambahkan dengan total yang sebelumnya sudah di inisialisasi dengan nilai 0 dan mencetak total saat ini, jika kondisi sudah bernilai False maka akan mencetak total keseluruhan

1.2.5 Soal 5

Buatlah flowchart dan program untuk membaca bilangan integer sampai user memasukkan angka = -100, maka program akan berhenti. Dan jumlahkan serta cari mean bilangan positif yang telah diinput !



```

1 total = 0
2 jml_bil = 0
3 while True:
4     bil = int(input('Masukkan Bilangan'))
5     if (bil == -100):
6         break
7     print(f'Operasikan {total} + {bil}')
8     jml_bil += 1
9     total += bil
10    mean = total / jml_bil
11
12    print(f'Jumlah saat ini = {total}')
13    print(f'Jumlah bilangan saat ini = {jml_bil}')
14    print(
15        '-----')
16    print(f'Jumlah bilangan = {total}')
17    print(f'Rata-rata bilangan = {mean}')

```

Output :

```

-----
Operasikan 35 + 12
Jumlah saat ini = 47
Jumlah bilangan saat ini = 4
-----
Operasikan 47 + 13
Jumlah saat ini = 60
Jumlah bilangan saat ini = 5
-----
Jumlah bilangan = 60
Rata-rata bilangan = 12.0

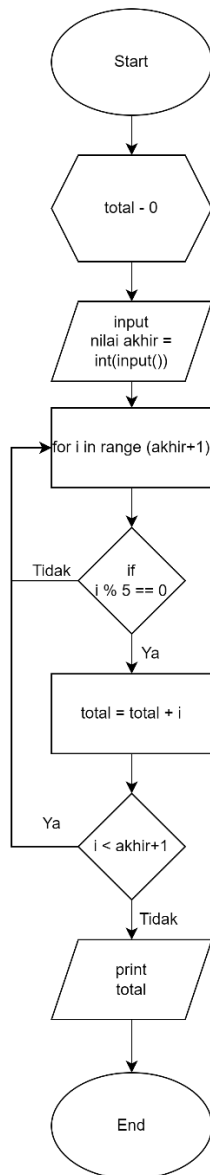
```

Penjelasan :

Dalam program ini pertama-tama kita menginisiasi nilai total dan jumlah bilangan adalah 0, lalu masuk ke perulangan while dengan perulangan tanpa batas dengan syarat angka yang di input oleh user tidak sama dengan -100, jika iya maka perulangan while akan berhenti dan menampilkan output berupa print total dan rata rata, jika tidak maka kode akan terus mengalami perulangan dan total akan terus bertambah begitu juga dengan jumlah bilangan nya, serta menampilkan total saat itu dan jumlah bilangan yang telah di inputkan

1.2.6 Soal 6

Buatlah flowchart dan program untuk membaca bilangan integer dan menjumlahkan semua bilangan kelipatan 5!



```

1 akhir = int(input('Masukkan batas akhir angka : '))
2 total = 0
3 for i in range(akhir+1) :
4     if i % 5 == 0 :
5         total += i
6
7 print(f'Total bilangan kelipatan 5 antara 1 sd {akhir}
  = {total}')
  
```

Output :

Total bilangan kelipatan 5 antara 1 sd 4 = 0

Total bilangan kelipatan 5 antara 1 sd 5 = 5

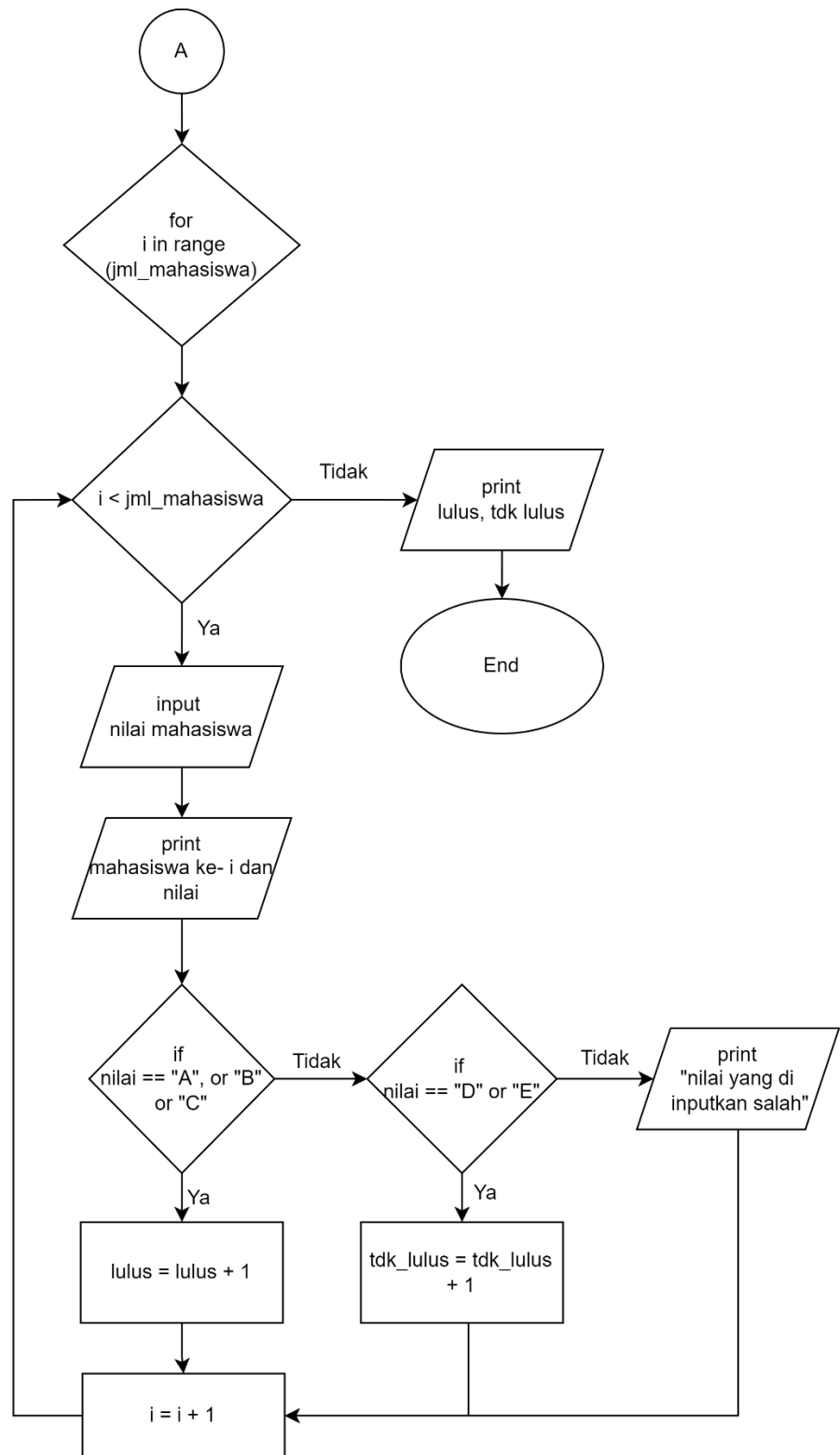
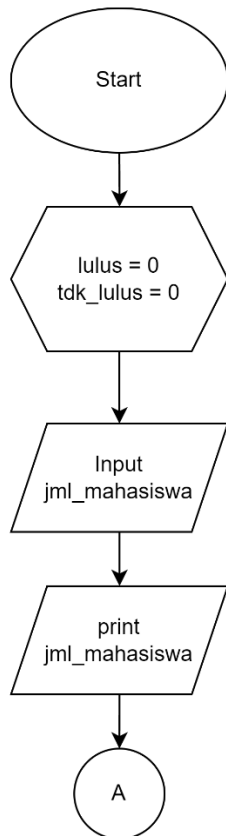
Total bilangan kelipatan 5 antara 1 sd 26 = 75

Penjelasan :

Dalam program ini user diminta untuk menentukan batas perulangan for dengan memasukkan input batas akhir angka, lalu setelah batas akhir di dapatkan, batas tersebut, menjadi batas index dari perulangan for, didalam perulangan tersebut, jika $i \% 5 = 0$, maka total di increment, setelah itu pada akhir program akan di print total penjumlahan bilangan kelipatan 5 antara 1 sd batas akhir

1.2.7 Soal 7

Buatlah flowchart dan program untuk membaca nilai mahasiswa dengan kategori: "A", "B", "C" dan dinyatakan lulus. Sedangkan kategori: "D", "E" tidak lulus kuliah. Serta tampilkan berapa banyak mahasiswa yang lulus dan tidak lulus



```

1 lulus = 0
2 tdk_lulus = 0
3 jumlah_mhs = int(input('Masukkan jumlah mahasiswa : '))
4 print(f'Jumlah mahasiswa = {jumlah_mhs}')
5
6 for i in range(jumlah_mhs) :
7     nilai = input('Masukkan nilai mahasiswa : ').upper()
8     print(f'Nilai mahasiswa ke-{i+1} adalah : {nilai}')
9     if nilai == 'A' or nilai == 'B' or nilai == 'C':
10         lulus += 1
11     elif nilai == 'D' or nilai == 'E':
12         tdk_lulus += 1
13     else:
14         print(f'Nilai yang anda inputkan pada mahasiswa ke-{i+1}
15             Salah!!!')
16
17 print(f'Jumlah mahasiswa lulus = {lulus}')
18 print(f'Jumlah mahasiswa tidak lulus = {tdk_lulus}')

```

```

Jumlah mahasiswa = 5
Nilai mahasiswa ke-1 adalah : A
Nilai mahasiswa ke-2 adalah : B
Nilai mahasiswa ke-3 adalah : C
Nilai mahasiswa ke-4 adalah : D
Nilai mahasiswa ke-5 adalah : E
Jumlah mahasiswa lulus = 3
Jumlah mahasiswa tidak lulus = 2

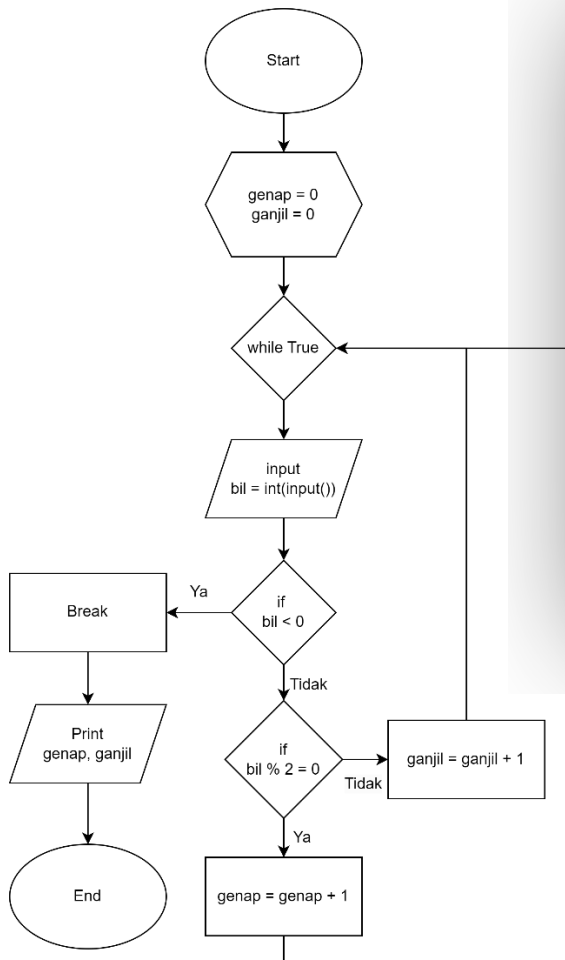
```

Penjelasan :

Pada program ini awal awal kita menginisialisasi nilai lulus dan tdk lulus menjadi 0, lalu kita meminta user untuk menginputkan jumlah mahasiswa, dan menampilkan jumlah mahasiswa, lalu kita masuk ke perulangan for dengan perulangan sebanyak jumlah mahasiswa, selama index tidak lebih atau sama dari jumlah mahasiswa, user di minta memasukkan nilai mahasiswa, jika nilai nya A, B, C maka mahasiswa di nyatakan lulus dan nilai dari lulus dan nilai lulus adalah $\text{lulus} = \text{lulus} + 1$, jika nilai nya D atau E, maka mahasiswa tidak dinyatakan lulus dan nilai dari tdk_lulus adalah $\text{tdk_lulus} = \text{tdk_lulus} + 1$, jika nilai tidak sama dari A,BC,D,E maka akan mengeluarkan print "Nilai yang diinputkan salah", setelah perulangan selesai akan di tampilkan jumlah atau nilai akhir dari lulus dan tdk_lulus

1.2.8 Soal 8

Buatlah flowchart dan program untuk membaca sekumpulan bilangan bulat positif. Pembacaan data diakhiri jika pengguna memasukkan nilai negatif. Selanjutnya, cetaklah berapa banyak bilangan genap dan ganjil.



```

1 genap = 0
2 ganjil = 0
3 while True:
4     bil = int(input('Masukkan bilangan bulat'))
5
6     if bil < 0:
7         break
8     else :
9         if bil%2 == 0:
10            genap += 1
11        else:
12            ganjil += 1
13
14 print(f'Jumlah bilangan Genap = {genap}')
15 print(f'Jumlah bilangan Ganjil = {ganjil}')

```

Output :

Jika inputan adalah
1, 2, 4, 4, 5, 5, 6, -1

Jumlah bilangan Genap = 4
Jumlah bilangan Ganjil = 3

Penjelasan :

Pada program ini, awal awal kita menginisialisasi nilai ganjil dan genap dengan nol, lalu kita masuk ke perulangan while True, artinya perulangan tanpa batas, dalam perulangan tersebut user di minta untuk memasukkan sebuah bilangan, jika bilangannya lebih dari 0, maka akan di lakukan pengecekan apakah $bil \% 2 = 0$, jika iya maka $genap = genap + 1$, jika tidak maka $ganjil = ganjil + 1$, perulangan akan terus berjalan sampai user memasukkan bilangan kurang dari 0, jika user memasukkan bilangan kurang dari 0 maka perulangan akan berhenti dan akan menampilkan print nilai akhir genap dan ganjil

1.3. Kesimpulan dan Saran

1.3.1. Kesimpulan

Dari praktikum repetition menggunakan python ini dapat disimpulkan :

1. bahasa pemrograman python memiliki 2 jenis perulangan yaitu while dan for
2. for loop di gunakan ketika kita tahu ingin berapa kali mengulang, misal 5-10 kali, atau berdasarkan inputan user dan berhenti ketika index sudah mencapai batas perulangan

3. while di gunakan ketika kita ingin melakukan perulangan dengan kondisi tertentu, pada perulangan while, kita perlu membuat suatu kondisi untuk memberhentikan perulangan, atau bisa menggunakan break.
4. Index dalam perulangan for secara default adalah di mulai dari 0
5. Dengan melakukan perulangan kita jadi bisa membuat kode lebih ringkas dan juga membuat kode program yang kita buat menjadi lebih efektif
6. Kita bisa menggunakan "continue" untuk melewati suatu index dalam perulangan dan "break" untuk memberhentikan perulangan pada index tertentu

1.3.2. Saran

Saran dan hal hal yang harus di perhatikan terhadap praktikum kali ini yaitu:

1. Pastikan teliti saat melakukan perhitungan index, karena secara default index dimulai dari 0
2. Tidak seperti for, dalam while kita perlu membuat kondisi tertentu agak menghindari infinite loop, kita bisa membuat kondisi false, atau melakukan break
3. Pastikan indentasi harus sesuai agar kita bisa menentukan dengan benar mana kode yang di jalankan didalam perulangan, dan mana kode yang di jalankan di luar perulangan
4. Hindari melakukan inisialisasi variabel di dalam perulangan karna variabel tersebut akan selalu te reset ke awal

DAFTAR PUSTAKA

1. Modul *Perkuliahan dan Praktikum Algoritma dan Pemrograman* Oleh Ilhamsyah, S.Si., M.Cs.
2. <https://www.w3schools.com/python/>