


<p><b>Nama:</b> Dave Ryano F.M</p> <p><b>NIM:</b> 064002300039</p>	 <p><b>Praktikum Algoritma &amp; Pemrograman</b></p>	<p><b>MODUL 4</b></p> <p><b>Nama Dosen:</b> Ratna Shofiati, S.Kom, M. Kom</p>
<p><b>Hari/Tanggal:</b> Kamis 12 Oktober 2023</p>		<p><b>Nama Asisten Labratorium:</b> 1. Yuda Hadi Prasetyo 065002100004 2. M.Hasan Husein 065002100009</p>

## Struktur Kendali II (Control Structure II)

### 1. Teori Singkat

#### Operasi Perulangan

Perulangan dalam bahasa pemrograman berfungsi menugaskan komputer untuk melakukan sesuatu secara berulang-ulang. Terdapat dua jenis perulangan dalam bahasa pemrograman python, yaitu perulangan dengan *while* dan *for*. Perulangan *for* disebut counted loop (perulangan yang terhitung), sementara perulangan *while* disebut uncounted loop (perulangan yang tak terhitung). Perbedaannya adalah perulangan *for* biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya. Sementara *while* untuk perulangan yang memiliki syarat dan tidak tentu berapa banyak perulangannya. Contohnya:



Berikut contoh program yang menggunakan perulangan *while*:

Source Code

```
    . . .  
  
    ulang_i = 0  
  
    while (ulang_i < 10):  
        print("Kuulang_i ", ulang_i, " kali")  
        ulang_i = ulang_i + 1
```

Output

```
    . . .  
  
    Kuulang_i      0      kali  
    Kuulang_i      1      kali  
    Kuulang_i      2      kali  
    Kuulang_i      3      kali  
    Kuulang_i      4      kali  
    Kuulang_i      5      kali  
    Kuulang_i      6      kali  
    Kuulang_i      7      kali  
    Kuulang_i      8      kali  
    Kuulang_i      9      kali
```



Berikut contoh program yang menggunakan perulangan *for*:

#### Source Code

```

for i in range(5):
    print("Nilai i adalah ", i)

for i in range(3,10):
    print("Kuulangi sebanyak ", i,
        "kali")

for i in range(30,20,-2):
    print("Aku mundur dari 30 ", i)

```

#### Output

```

Nilai i adalah 0
Nilai i adalah 1
Nilai i adalah 2
Nilai i adalah 3
Nilai i adalah 4
Kuulangi sebanyak 3 kali
Kuulangi sebanyak 4 kali
Kuulangi sebanyak 5 kali
Kuulangi sebanyak 6 kali
Kuulangi sebanyak 7 kali
Kuulangi sebanyak 8 kali
Kuulangi sebanyak 9 kali
Aku mundur dari 30 30
Aku mundur dari 30 28
Aku mundur dari 30 26
Aku mundur dari 30 24
Aku mundur dari 30 22

```



## 2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)

## 3. Elemen Kompetensi

### a. Latihan pertama

Buatlah program dengan keluaran output seperti dibawah ini menggunakan perulangan *for*

7777777

666666

55555

4444

333

22

1

### Source Code

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Thu Oct 12 19:43:12 2023
4
5  @author: Oaan115
6  """
7
8  for i in range (1):
9      print("7777777")
10 for i in range (1):
11     print("666666")
12 for i in range (1):
13     print("55555")
14 for i in range (1):
15     print("4444")
16 for i in range (1):
17     print("333")
18 for i in range (1):
19     print("22")
20 for i in range (1):
21     print ("1")
```



## Output

```
C:\Users\Oaan115\Documents\Python Scripts>python segitiga2.py
7777777
6666666
55555
4444
333
22
1

C:\Users\Oaan115\Documents\Python Scripts>python segitiga2.py
7777777
6666666
55555
4444
333
22
1
```

### b. Latihan Kedua

Buatlah program yang sebelumnya telah dibuat yang menentukan jumlah hari dalam suatu bulan sesuai dengan inputan bulan dan tahun yang diinputkan oleh user menggunakan implementasi perulangan *while*.

## Source Code

```
bulan = int(input(" pilih bulannya (1-12): "))
tahun = int(input(" pilih tahunnya (misalnya 2023): "))

bulan_hari = {
    1: 31,
    2: 28 if (tahun % 4 == 0 and tahun % 100 != 0) or (tahun % 400 == 0) else
    3: 31,
    4: 30,
    5: 31,
    6: 30,
    7: 31,
    8: 31,
    9: 30,
    10: 31,
    11: 30,
    12: 31\
}

if bulan < 1 or bulan > 12:
    print("Bulan yang diinput tidak valid.")
else:
    print(f"Jumlah hari dalam bulan {bulan} tahun {tahun} adalah {bulan_hari[b"]
```



### Output

```
C:\Users\Oaan115\Documents\Python Scripts>python bulan.py
Masukkan bulan (1-12): 2
Masukkan tahun (misalnya 2023): 2008
Jumlah hari dalam bulan 2 tahun 2008 adalah 28 hari.

C:\Users\Oaan115\Documents\Python Scripts>python bulan.py
Masukkan bulan (1-12): 2
Masukkan tahun (misalnya 2023): 2009
Jumlah hari dalam bulan 2 tahun 2009 adalah 29 hari.

C:\Users\Oaan115\Documents\Python Scripts>python bulan.py
Masukkan bulan (1-12): 3
Masukkan tahun (misalnya 2023): 3011
Jumlah hari dalam bulan 3 tahun 3011 adalah 31 hari.

C:\Users\Oaan115\Documents\Python Scripts>python bulan.py
Masukkan bulan (1-12): 6
Masukkan tahun (misalnya 2023): 2005
Jumlah hari dalam bulan 6 tahun 2005 adalah 30 hari.

C:\Users\Oaan115\Documents\Python Scripts>python bulan.py
Masukkan bulan (1-12): 12
Masukkan tahun (misalnya 2023): 2030
Jumlah hari dalam bulan 12 tahun 2030 adalah 31 hari.
```

#### 4. File Praktikum

Github Repository:

```
print("Taruh link github repository praktikum kalian disini yaa...")
```

#### 5. Soal Latihan

Soal:

1. Dalam perulangan while kondisi apakah yang menyebabkan *infinite loop*? Jelaskan dan berikan contohnya!
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. sebuah infinite loop terjadi ketika kondisi yang diberikan selalu benar. Ini berarti perulangan akan terus berulang tanpa akhir sampai kondisinya berubah menjadi salah (False) atau henti secara manual.

Contoh infinite loop dalam Python:

```
while True:
```

```
    print("Ini adalah infinite loop")
```



2. Program dimulai dengan memasukkan bulan dan tahun yang ingin diperiksa jumlah hari dalamnya. Input bulan disimpan dalam variabel `bulan` dan input tahun disimpan dalam variabel `tahun`. Kemudian, program menggunakan perulangan `while` untuk memeriksa apakah input bulan valid atau tidak. Setelah nilai bulan valid ditemukan, program memeriksa apakah tahun tersebut adalah tahun kabisat atau bukan. Sebuah tahun kabisat memiliki 29 hari pada bulan Februari, sedangkan tahun biasa hanya memiliki 28 hari pada bulan Februari. Program menghitung jumlah hari dalam bulan tersebut berdasarkan bulan dan tahun yang diberikan oleh pengguna. Ini dapat dilakukan dengan menggunakan pernyataan kondisional `if-elif-else` untuk masing-masing bulan, dengan mempertimbangkan apakah itu bulan Februari pada tahun kabisat atau tahun biasa. Terakhir, program mencetak jumlah hari dalam bulan yang sesuai berdasarkan inputan pengguna.

## 6. Kesimpulan

- a. Dalam pengerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
- b. Kita dapat mengetahui bahwa kita dapat membuat program untuk membuat sistem operasi pengulangan di python dan juga membuat program dalam menentukan jumlah hari dalam suatu bulan sesuai dengan inputan bulan dan tahun menggunakan while dan bahasa lainnya

## 7. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	✓	
2.	Latihan Kedua	✓	



## 8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	30 Menit	Menarik
2.	Latihan Kedua	30 Menit	Menarik

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang

