

<p>Nama: Steven Raja P M</p> <p>NIM: 064102400018</p>	 <p>Praktikum Algoritma & Pemrograman</p>	<h1>MODUL 6</h1> <p>Nama Dosen: Binti solihah, S.T, M.KOM</p> <p>Nama Asisten Labratorium:</p> <ol style="list-style-type: none">Yustianas Rombon - 064002300015Vira Aditya Kurniawan - 065002300012
---	---	--

Fungsi (Function) pada Pemrograman Python

1. Teori Singkat

Fungsi

Fungsi merupakan suatu bagian dari program yang dimaksudkan untuk mengerjakan suatu tugas tertentu dan letaknya terpisah dari program yang memanggilnya. Pada pembuatan program yang kompleks dan memiliki banyak fitur, kita diharuskan menggunakan fungsi. Fungsi diperlukan agar mempermudah kita dalam membaca sebuah kode program dan mempermudah untuk merawatnya. Dengan adanya sebuah fungsi kita tidak perlu menulis kode sepanjang gerbang kereta api di program utama dan kita dapat memecah atau membaginya lalu tinggal memanggil fungsinya saja didalam program utama. Dan itu juga akan mempersingkat penulisan dari program utama tersebut nantinya

Fungsi memiliki parameter yang dapat dipergunakan untuk memasukkan atau menampung variabel kedalam sebuah fungsi. Dalam pendeklarasian parameter pada sebuah fungsi kita juga dapat memasukkan *default argument/parameter*.

Fungsi juga dapat mengembalikan nilai dengan cara menggunakan keyword *return* yang di taruh didalam blok program fungsi itu sendiri untuk mengembalikan nilai yang ingin kita kembalikan kedalam program utama.

Membuat Fungsi pada Python

Fungsi pada Python, dibuat dengan kata kunci *def* kemudian diikuti dengan nama fungsinya.



```
# Pendeklarasian sebuah fungsi
def nama_fungsi():
    print("Hello Ini Fungsi")

# Pemanggilan sebuah fungsi
nama_fungsi()
```

Fungsi dengan parameter pada Python

Memasukkan parameter kedalam sebuah fungsi

```
# pendeklarasian fungsi
def luas_persegi(sisi):
    luas = sisi * sisi
    return luas

# pemanggilan fungsi
print("Luas persegi: ", luas_persegi(6))
```

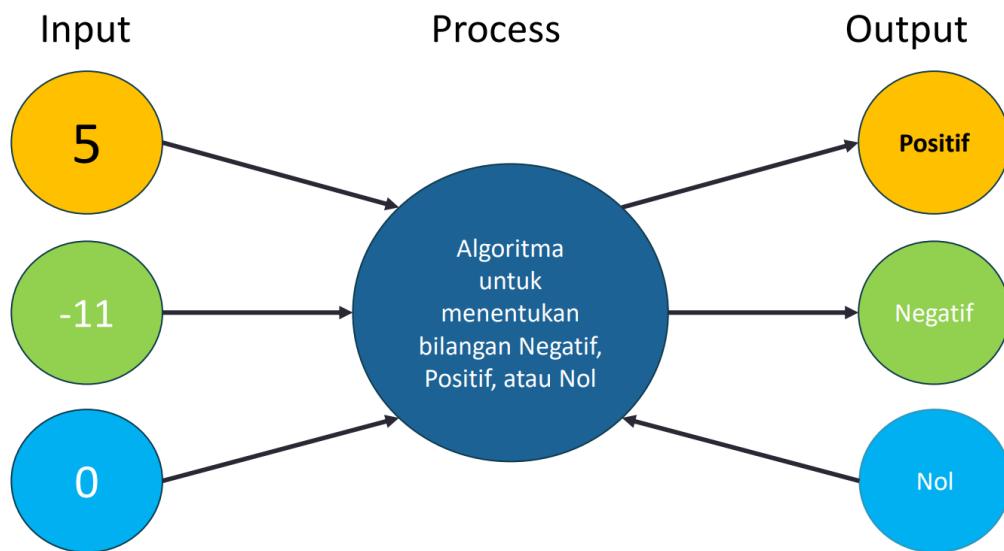


IPO (Input Process Output)

Konsep Dasar Input, Process, dan Output (IPO)

- Konsep input, process, dan output adalah prinsip dasar dalam pemrograman dan pengembangan algoritma.
- Setiap algoritma melibatkan tiga tahap utama: mengambil data masukan (input), melakukan operasi atau pengolahan data (process), dan menghasilkan hasil akhir (output).
- Konsep ini menggambarkan bagaimana algoritma beroperasi untuk memproses informasi.

Gambaran IPO (Menentukan Bilangan)



Notasi Algoritma Flowchart

1. Flowchart adalah representasi visual atau diagram alir yang digunakan untuk menggambarkan langkah-langkah dan urutan proses suatu algoritma atau program.
2. Flowchart menyajikan langkah-langkah dalam bentuk simbol-simbol grafis yang saling terhubung, membantu dalam memvisualisasikan bagaimana informasi mengalir dan bagaimana proses dilakukan.
3. Dalam kaitannya dengan notasi deskriptif, notasi algoritma yang menggunakan flowchart dapat lebih cepat dibaca dan dilihat alur dan hubungannya.

Simbol-simbol pada Flowchart



- Setiap elemen flowchart dihubungkan oleh garis aliran bertanda panah
- Garis aliran dimulai dari atas symbol dan keluar dari bagian bawah, kecuali symbol keputusan yang alirannya keluar dari bawah atau samping
- Aliran bergerak dari atas ke bawah
- Proses awal dan akhir menggunakan symbol terminal.

... Simbol-simbol pada Flowchart



Terminator yang menandakan *Start* (awal) atau *End* (akhir) program.



Flow line yang digunakan untuk menunjukkan arah aliran pada program.



Process menunjukkan proses yang dilakukan pada masukan.



Input atau output untuk menunjukkan masukan dan keluaran.



Preparation digunakan untuk membuat deklarasi nilai awal.



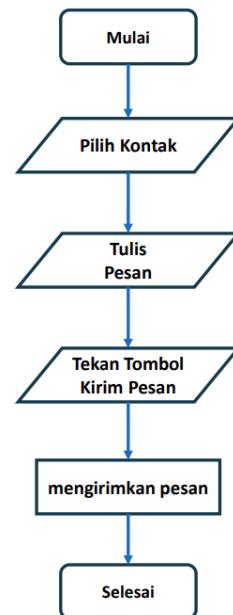
On Page Connector digunakan untuk menghubungkan antar *flowchart*



Decision menunjukkan keputusan atau kondisi untuk memilih keputusan.

Contoh sederhana Penggunaan *flowchart* untuk menunjukkan algoritma

Kasus/Aliran:
Mengirim pesan WhatsApp



2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)

3. Elemen Kompetensi



a. Latihan pertama

Buatlah program yang sebelumnya telah dibuat yaitu program untuk merata-ratakan nilai sesuai dengan kategori huruf yang diinputkan dengan mengimplementasikan fungsi yang sudah dipelajari. Persyaratan program yaitu fungsi menggunakan *default argument/parameter* dan mengimplementasikan pengembalian nilai.

IPO (Input Process Output)

```
def hitung_rata_rata(nilai=0, counter=0):
    while True:
        implementasi = input("Masukkan grade (atau ketik 'STOP' untuk selesai): ")

        if implementasi == "A":
            counter += 1
            nilai += 4.00
        elif implementasi == "A-":
            counter += 1
            nilai += 3.75
        elif implementasi == "B+":
            counter += 1
            nilai += 3.50
        elif implementasi == "B":
            counter += 1
            nilai += 3.00
        elif implementasi == "B-":
            counter += 1
            nilai += 2.75
        elif implementasi == "C+":
            counter += 1
            nilai += 2.50
        elif implementasi == "C":
            counter += 1
            nilai += 2.00
        elif implementasi == "C-":
            counter += 1
            nilai += 1.75
        elif implementasi == "D":
            counter += 1
            nilai += 1.50
        elif implementasi == "E":
            counter += 1
```



```

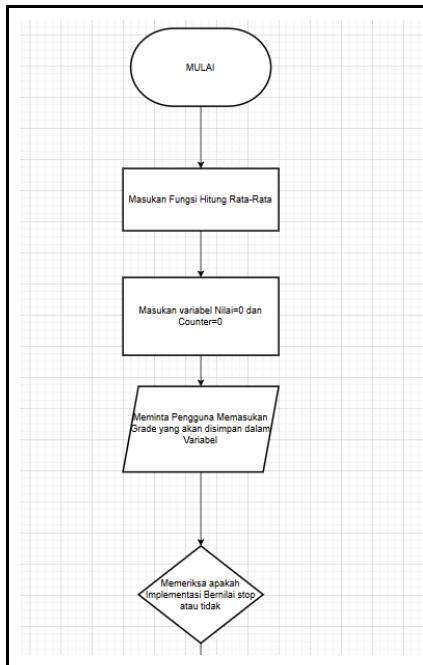
        nilai += 1.00
    elif implementasi == "STOP":
        if counter == 0:
            return 0
        return nilai / float(counter)
    else:
        print("Grade tidak valid. Masukkan grade yang benar.")

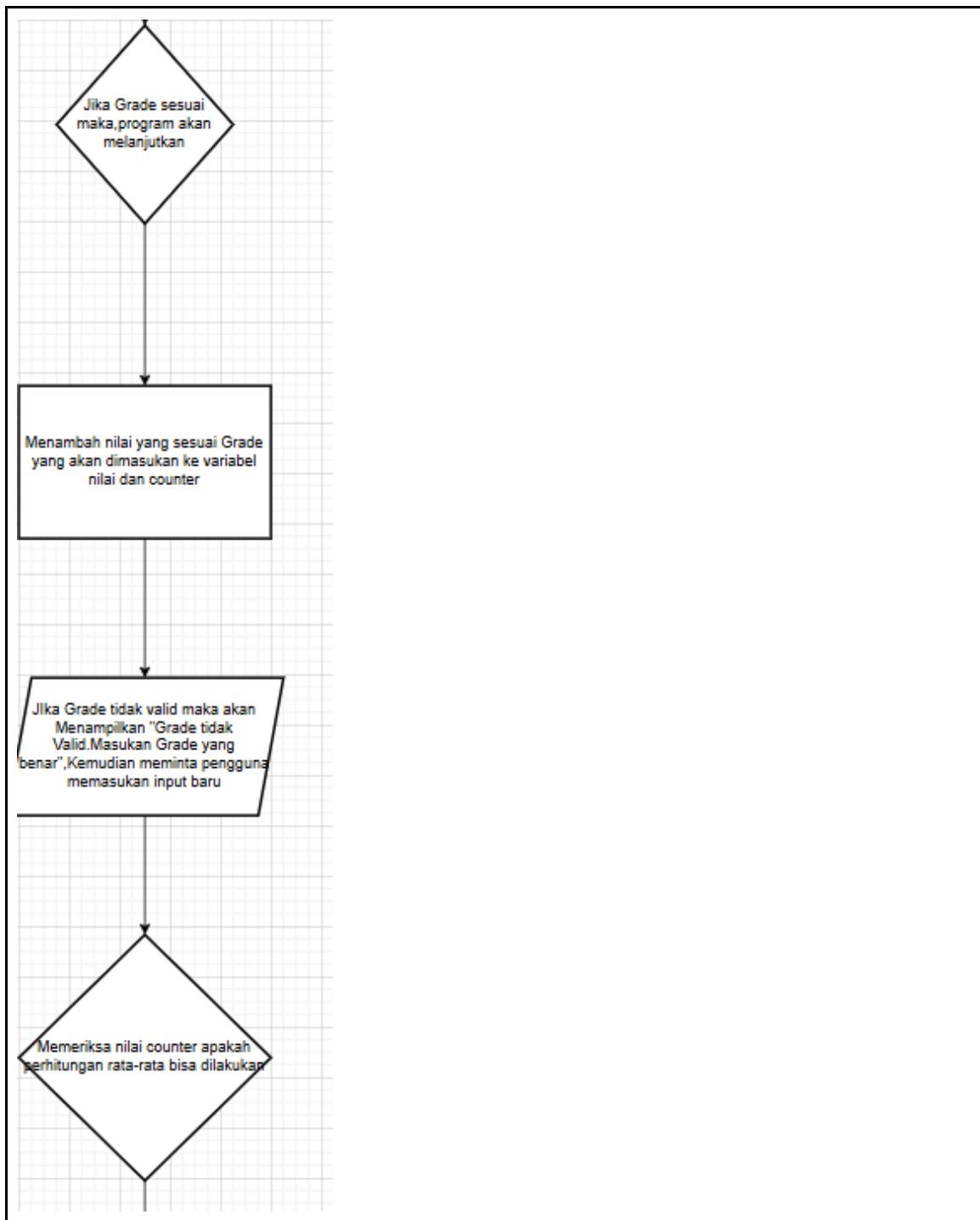
rata_rata = hitung_rata_rata()
print("Nilai rata-rata:", rata_rata)

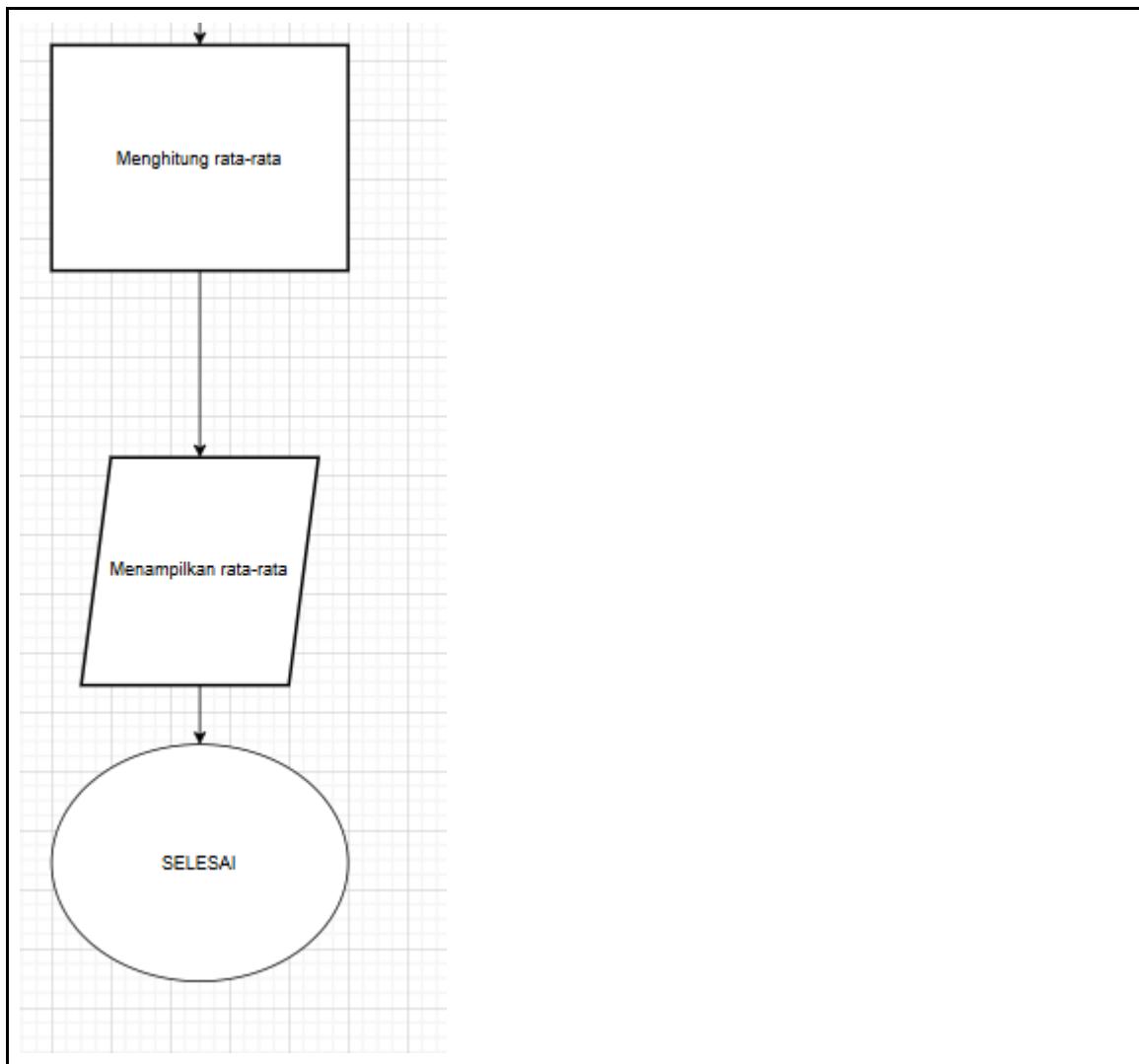
→ Masukkan grade (atau ketik 'STOP' untuk selesai): A
Masukkan grade (atau ketik 'STOP' untuk selesai): C
Masukkan grade (atau ketik 'STOP' untuk selesai): C-
Masukkan grade (atau ketik 'STOP' untuk selesai): STOP
Nilai rata-rata: 2.5833333333333335

```

Flowchart







Source Code



```
def hitung_rata_rata(nilai=0, counter=0):
    while True:
        implementasi = input("Masukkan grade (atau ketik 'STOP' untuk selesai): ")

        if implementasi == "A":
            counter += 1
            nilai += 4.00
        elif implementasi == "A-":
            counter += 1
            nilai += 3.75
        elif implementasi == "B+":
            counter += 1
            nilai += 3.50
        elif implementasi == "B":
            counter += 1
            nilai += 3.00
        elif implementasi == "B-":
            counter += 1
            nilai += 2.75
        elif implementasi == "C+":
            counter += 1
            nilai += 2.50
        elif implementasi == "C":
            counter += 1
            nilai += 2.00
        elif implementasi == "C-":
            counter += 1
            nilai += 1.75
        elif implementasi == "D":
            counter += 1
            nilai += 1.50
        elif implementasi == "E":
            counter += 1
            nilai += 1.00
        elif implementasi == "STOP":
            if counter == 0:
                return 0
            return nilai / float(counter)
        else:
            print("Grade tidak valid. Masukkan grade yang benar.")
```



```
rata_rata = hitung_rata_rata()  
print("Nilai rata-rata:", rata_rata)
```

Output

→ Masukkan grade (atau ketik 'STOP' untuk selesai): A
Masukkan grade (atau ketik 'STOP' untuk selesai): C
Masukkan grade (atau ketik 'STOP' untuk selesai): C-
Masukkan grade (atau ketik 'STOP' untuk selesai): STOP
Nilai rata-rata: 2.5833333333333335

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 dengan mengimplementasikannya menggunakan fungsi termasuk memperhatikan tahun kabisat dan non kabisat. Gunakan 2 fungsi beserta implementasikan parameternya.

IPO (input process output)



```
def is_kabisat(tahun):
    return tahun % 4 == 0 and (tahun % 100 != 0 or tahun % 400 == 0)

def jumlah_hari(bulan, tahun):
    if bulan == 2:
        return 29 if is_kabisat(tahun) else 28
    elif bulan in [4, 6, 9, 11]:
        return 30
    else:
        return 31

while True:
    bulan = int(input("Masukkan bulan (1-12): "))
    tahun = int(input("Masukkan tahun: "))

    hari = jumlah_hari(bulan, tahun)
    print(f"Jumlah hari pada bulan {bulan} tahun {tahun} adalah {hari} hari.")

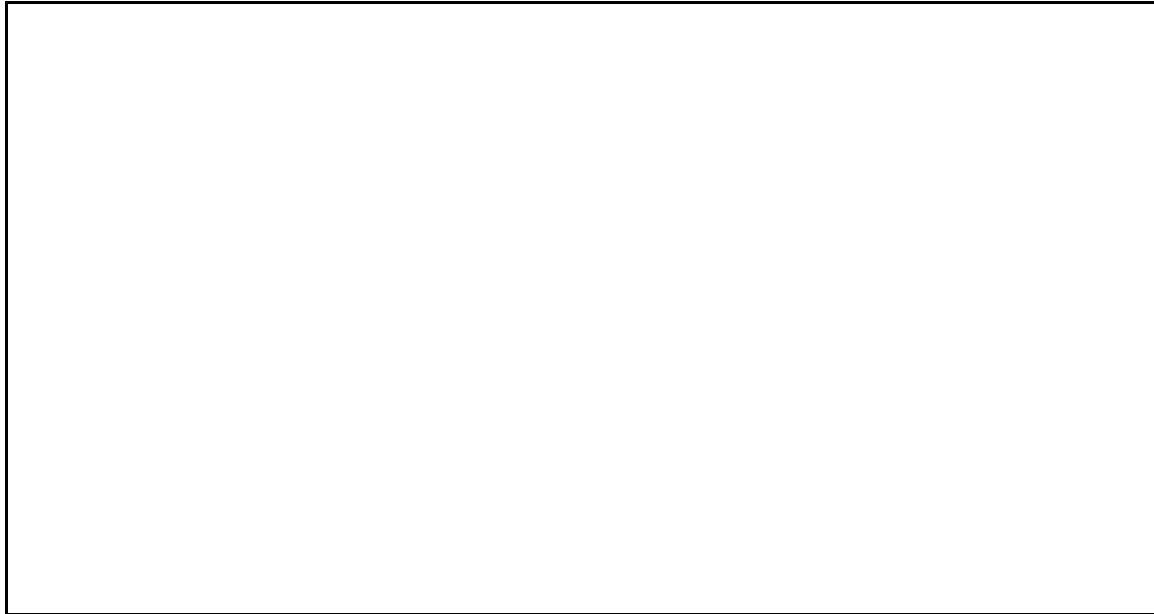
    if is_kabisat(tahun):
        print(f"{tahun} adalah tahun kabisat.")

    lanjut = input("Ingin memasukkan bulan dan tahun lain? (y/n): ").lower()
    if lanjut != 'y':
        print("Program selesai.")
        break
```

→ Masukkan bulan (1-12): 11
Masukkan tahun: 2005
Jumlah hari pada bulan 11 tahun 2005 adalah 30 hari.
Ingin memasukkan bulan dan tahun lain? (y/n): y
Masukkan bulan (1-12): 2
Masukkan tahun: 2024
Jumlah hari pada bulan 2 tahun 2024 adalah 29 hari.
2024 adalah tahun kabisat.
Ingin memasukkan bulan dan tahun lain? (y/n): n
Program selesai.



Flowchart



Source Code

```
def is_kabisat(tahun):
    return tahun % 4 == 0 and (tahun % 100 != 0 or tahun % 400 == 0)

def jumlah_hari(bulan, tahun):
    if bulan == 2:
        return 29 if is_kabisat(tahun) else 28
    elif bulan in [4, 6, 9, 11]:
        return 30
    else:
        return 31

while True:
    bulan = int(input("Masukkan bulan (1-12): "))
    tahun = int(input("Masukkan tahun: "))

    hari = jumlah_hari(bulan, tahun)
    print(f"Jumlah hari pada bulan {bulan} tahun {tahun} adalah {hari} hari.")
```



```
if is_kabisat(tahun):
    print(f"{tahun} adalah tahun kabisat.")

lanjut = input("Ingin memasukkan bulan dan tahun lain? (y/n): ")
.lower()
if lanjut != 'y':
    print("Program selesai.")
    break
```

Output

```
→ Masukkan bulan (1-12): 11
Masukkan tahun: 2005
Jumlah hari pada bulan 11 tahun 2005 adalah 30 hari.
Ingin memasukkan bulan dan tahun lain? (y/n): y
Masukkan bulan (1-12): 2
Masukkan tahun: 2024
Jumlah hari pada bulan 2 tahun 2024 adalah 29 hari.
2024 adalah tahun kabisat.
Ingin memasukkan bulan dan tahun lain? (y/n): n
Program selesai.
```

4. File Praktikum

Github Repository:

<https://github.com/Stevenraja/PrakAlgo6.git>



5. Soal Latihan

Soal:

1. Jelaskan fungsi utama default *argument/parameter* dalam fungsi dan mengapa perlu menggunakan *default parameter/argument* pada sebuah fungsi?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. mempermudah pemanggilan fungsi, meningkatkan fleksibilitas, mengurangi error, dan membuat kode lebih efisien serta mudah dibaca.
2. Masukan fungsi `is_kabisat(tahun)` untuk menerima parameter tahun dan mengembalikan true jika benar false jika bukan, fungsi `jumlah_hari(bulan,tahun)` untuk mengembalikan jumlah hari dalam bulan, Loop while true untuk memasukan bulan dan tahun secara berulang sampai ingin diberhentikan.

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 Fungsi Fungsi dalam bahasa python dan bagaimana cara menjalankannya dengan contoh fungsi `is_kabisat(tahun)` dan fungsi `jumlah_hari(bulan,tahun)` dan bagaimana cara looping pada fungsi.

7. Cek List (✓)

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

8. Formulir Umpam Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	40 Menit	Menarik
2.	Latihan Kedua	40 Menit	Menarik



Keterangan:

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

