



---

# Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

NIM	71251193
Nama Lengkap	Cheila Zefanya Wibowo
Minggu ke / Materi	04/ Struktur Kontrol Percabangan

PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA  
YOGYAKARTA  
2026

## BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

### Boolean Expression dan Logical Operator

Boolean expression adalah nilai yang menghasilkan True(benar) atau False(salah). Simbol yang berupa kata dan digunakan untuk menjalankan operasi logika. Berikut contoh perintah untuk melihat hasil nilai Boolean expression.

```
(base) C:\Users\LENOVO>python
Python 3.13.9 | packaged by Anaconda, Inc. | (main, Oct 21 2025, 19:09:58) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> pembelian = 80000
>>> pembelian >= 50000
True
>>> pembelian = 10000
>>> pembelian >= 50000
False
>>> pembelian = 300000
>>> pembelian >= 100000
True
>>> a = 'cheila'
>>> c = 'zefanya'
>>> a is c
False
>>> a is not c
True
>>> a = 28
>>> b = 19
>>> a is b
False
>>> a is not b
True
>>> |
```

Gambar 1.1 : contoh penggunaan Boolean expression dalam anaconda prompt.

Boolean expression dapat juga disusun menggunakan operator-operator seperti pada gambar 1.2.

Operator	Keterangan
x == y	Apakah x sama dengan y?
x != y	Apakah x tidak sama dengan y?
x > y	Apakah x lebih besar dari y?
x >= y	Apakah x lebih besar atau sama dengan y?
x < y	Apakah x lebih kecil dari y?
x <= y	Apakah x lebih kecil atau sama dengan y?
x is y	Apakah x sama dengan y?
x is not y	Apakah x tidak sama dengan y?

Gambar 1.2 : operator perbandingan (comparison).

Kita harus bisa menyusun bentuk boolean expression dan memilih operator yang tepat untuk masalah yang ada. Ada beberapa hal yang harus diperhatikan dalam menyusun boolean expression :

1. Bentuk boolean expression hanya ada dua, yaitu **True dan False**.
2. Teliti dan menentukan variabel yang perlu dibandingkan dengan benar sesuai masalah.
3. Memperhatikan kata-kata khusus seperti **tidak berbeda, tidak lebih dari, dsb**.

## Logika Operator

Operator logika adalah simbol yang dipakai untuk menghubungkan rumus-rumus logika. Operator logika juga bisa disebut sebagai kata kunci yang digunakan dalam pemrograman seperti AND, OR, NOT.

Berikut contoh kasus menggunakan logika operator

```
[4]: nilai = int(input("masukkan nilai: "))
    kehadiran = int(input("masukkan presentase kehadiran: "))

    #proses
    if nilai >= 80 and kehadiran >= 75:
        print("kamu lulus")
    else:
        print("semangat kamu belum lulus")

    masukkan nilai: 90
    masukkan presentase kehadiran: 70
    semangat kamu belum lulus
```

Gambar 1.3 : contoh penggunaan logika operator

```
[8]: total_belanja = int(input("masukkan total belanja: "))
    member = input("apakah memiliki member? (ya/tidak) : ")

    potongan_harga = total_belanja >= 150000 or member

    if potongan_harga:
        diskon = total_belanja * 0.20
        pembayaran = total_belanja - diskon
        print("selamat anda mendapatkan diskon")
    else:
        diskon = 0
        pembayaran = total_belanja
        print("maaf anda belum mendapatkan diskon")
    print("total yang dibayar: Rp", pembayaran)

    masukkan total belanja: 300000
    apakah memiliki member? (ya/tidak) : tidak
    selamat anda mendapatkan diskon
    total yang dibayar: Rp 240000.0
```

Gambar 1.4 : contoh penggunaan logika operator dengan kasus yang berbeda.

## Bentuk-bentuk percabangan

Percabangan pada python digunakan untuk mengambil Keputusan dalam kondisi tertentu dengan memiliki 3 bentuk, yaitu **if**, **if-else**, **if-else,elif**

1. If hanya dapat digunakan untuk kondisi Tunggal.

```
[16]: tahun_lahir = int(input("masukkan tahun lahir kamu: "))
    if 1997<= tahun_lahir <=2012 :
        print("kamu gen z")

    masukkan tahun lahir kamu: 2008
    kamu gen z
```

Gambar 1.5 : contoh penggunaan bentuk pertama (kondisi Tunggal).

2. If-else hanya dapat digunakan untuk dua kondisi

Percabangan dua kondisi hanya menghasilkan dua kondisi saja.

```
[17]: usia = int(input("masukkan usia anda: "))
    if 24 <= usia <= 10 :
        print("cie anak muda")
    else :
        print("maaf bukan anak muda lagi")

    masukkan usia anda: 25
    maaf bukan anak muda lagi
```

```
[23]: usia = int(input("masukkan usia anda: "))
      if 10 <= usia <= 24 :
          print("cie anak muda")
      else :
          print("maaf bukan anak muda")
```

```
masukkan usia anda: 11
cie anak muda
```

Gambar 1.6 : penggunaan bentuk kedua (dua kondisi)

### 3. If-else-elif digunakan untuk banyak kondisi

Pervabangan ini akan menghasilkan lebih dari dua kondisi.

```
[28]: #penilaian
      nilai = float(input("masukkan nilai kamu : "))
      if nilai >= 85 :
          print("nilai kamu A")
      elif nilai >= 75 :
          print("nilai kamu B")
      elif nilai >= 65 :
          print("nilai kamu c")
      else :
          print("nilai kamu d")
```

```
masukkan nilai kamu : 90
nilai kamu A
```

```
[29]: #penilaian
      nilai = float(input("masukkan nilai kamu : "))
      if nilai >= 85 :
          print("nilai kamu A")
      elif nilai >= 75 :
          print("nilai kamu B")
      elif nilai >= 65 :
          print("nilai kamu c")
      else :
          print("nilai kamu d")
```

```
masukkan nilai kamu : 75.5
nilai kamu B
```

```
[30]: #penilaian
      nilai = float(input("masukkan nilai kamu : "))
      if nilai >= 85 :
          print("nilai kamu A")
      elif nilai >= 75 :
          print("nilai kamu B")
      elif nilai >= 65 :
          print("nilai kamu c")
      else :
          print("nilai kamu d")

      masukkan nilai kamu : 55.5
      nilai kamu d
```

Gambar 1.7 : penggunaan bentuk ketiga (banyak kondisi)

## Penanganan Kesalahan Input Menggunakan Exception Handling

Saat menghadapi input yang salah dari pengguna, kita perlu memperhatikan kesalahan yang mungkin sering terjadi sehingga menyebabkan program tidak dapat berjalan. Biasanya pengguna langsung memasukkan saja, yang seharusnya memasukkan angka pengguna memasukkan huruf sehingga menyebabkan program tidak bisa berjalan.

```
[32]: usia = int(input("masukkan usia anda: "))
      if usia <= 5:
          print("Balita")
      elif usia >= 6 and usia <= 11:
          print("Kanak-Kanak")
      elif usia >= 12 and usia <= 25:
          print("Remaja")
      elif usia >= 26 and usia <= 45:
          print("Dewasa")
      elif usia > 45:
          print("Lansia")

      masukkan usia anda: 50
      Lansia
```

Gambar 1.8 : contoh penggunaan ketika input pengguna benar.

Jika pengguna memasukkan input dengan benar dan program akan mengeluarkan output sesuai yang kita inginkan.

Berikut contoh ketika pengguna salah memasukkan input yang digunakan

```
[33]: usia = int(input("masukkan usia anda: "))
      if usia <= 5:
          print("Balita")
      elif usia >= 6 and usia <= 11:
          print("Kanak-Kanak")
      elif usia >= 12 and usia <= 25:
          print("Remaja")
      elif usia >= 26 and usia <= 45:
          print("Dewasa")
      elif usia > 45:
          print("Lansia")

masukkan usia anda: dua puluh

Fix Code

-----
ValueError                                Traceback (most recent call last)
Cell In[33], line 1
----> 1 usia = int(input("masukkan usia anda: "))
      2 if usia <= 5:
      3     print("Balita")

ValueError: invalid literal for int() with base 10: 'dua puluh'
```

Gambar 1.9 : Ketika pengguna salah memasukan input

Nah, bagaimana menangani kasus yang begini? Solusi nya itu menggunakan **try** dan **except**. Ketika terdapat kesalahan input maka output program akan memberikan peringatan eror atau input salah.

```
[35]: try:
      usia = int(input("masukkan usia anda: "))
      if usia <= 5:
          print("Balita")
      elif usia >= 6 and usia <= 11:
          print("Kanak-Kanak")
      elif usia >= 12 and usia <= 25:
          print("Remaja")
      elif usia >= 26 and usia <= 45:
          print("Dewasa")
      elif usia > 45:
          print("Lansia")
      except:
          print("input salah!")

masukkan usia anda: dua lima
input salah!
```

Gambar 2.0 : contoh penggunaan try and except

Menggunakan **try** dan **except** ini mencegah untuk mencegah berhenti nya program secara tiba-tiba. **Try** dadalah bagian kode yang diuji kesalahan nya dan **except** untuk menangani kesalahan yang ada.

Sumber : Materi 04 Struktur Kontrol Percabangan (eclass)

## BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

Link GitHub : [https://github.com/chei-cell/71251193\\_cheila-.git](https://github.com/chei-cell/71251193_cheila-.git)

### SOAL 1

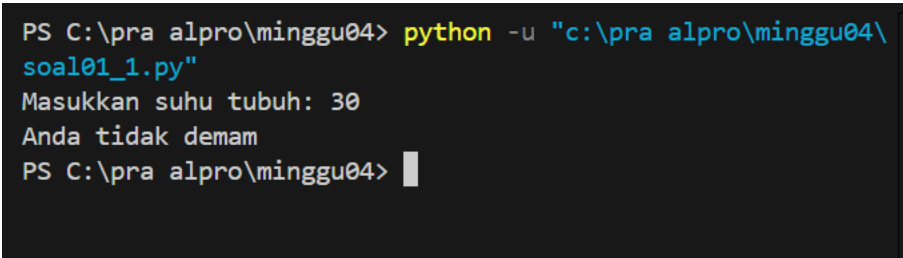
#### Source Code 4.1:

```
try:
    suhu = int(input("Masukkan suhu tubuh: "))
    if suhu >= 38:
        print("Anda demam")
    else:
        print("Anda tidak demam")
except:
    print("Input anda salah!")
```

#### Langkah-langkah :

1. Pada soal 3.1 program untuk menentukan suhu badan dengan mengeluarkan output keterangan demam atau tidak demam.
2. Maka dibutuhkan fungsi if else untuk mengetahui kedua kondisi tersebut.
3. Kita menggunakan **try-except** untuk menangani kesalahan input dari pengguna, jika pengguna salah memasukkan input maka akan muncul "Input anda salah!"

#### Output :



```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_1.py"
Masukkan suhu tubuh: 30
Anda tidak demam
PS C:\pra alpro\minggu04>
```



Output jika kita input suhu tubuh dibawah 38

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_1.py"
Masukkan suhu tubuh: 40
Anda demam
PS C:\pra alpro\minggu04>
```

Jika kita input >38

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_1.py"
Masukkan suhu tubuh: empat lima
Input anda salah!
PS C:\pra alpro\minggu04>
```

#### Source code 4.2:

```
try:

    bilangan = int(input("Masukkan bilangan: "))

    if bilangan > 0 :

        print("bilangan positif")

    elif bilangan < 0 :

        print("bilangan negatif")

    elif bilangan == 0:

        print("nol")

except:

    print("maaf input anda salah!")
```

#### Langkah-langkah :

1. Soal 4.2 diminta untuk menentukan apakah bilangan yang diinput termasuk bilangan positif, negatif, atau nol

2. Kita menggunakan fungsi `if elif` karena memiliki 3 kondisi.
3. Menggunakan **try – except** untuk menangani kesalahan input dari pengguna sehingga output yang dikeluarkan maaf input salah.

**Output:**

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_2.py"
Masukkan bilangan: 1
bilangan positif
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_2.py"
Masukkan bilangan: -1
bilangan negatif
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_2.py"
Masukkan bilangan: 0
nol
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal01_2.py"
Masukkan bilangan: enam
maaf input anda salah!
```

**Source Code 4.3:**

```
try:

    a = int(input("masukkan nilai a: "))

    b = int(input("masukkan nilai b: "))

    c = int(input("masukkan nilai c: "))

    if a > b and a > c :

        print("Terbesar: ", a)

    elif b > a and b > c :
```

```
        print("Terbesar: ", b)

    elif c > a and c > b :

        print("Terbesar: ",c)

except:

    print("input nya salah!")
```

### Langkah-langkah:

1. Soal 4.3 diminta untuk menentukan nilai terbesar dari 3 bilangan, maka pengguna diminta untuk input 3 bilangan terlebih dahulu.
2. Kita menggunakan if elif karena memiliki 3 kondisi.
3. Gunakan **try – except** untuk mengatasi kesalahan input dari pengguna.

### Output :

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\
tempCodeRunnerFile.py"
masukkan nilai a: 3
masukkan nilai b: 4
masukkan nilai c: 5
Terbesar: 5
PS C:\pra alpro\minggu04> █
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\
tempCodeRunnerFile.py"
masukkan nilai a: 4
masukkan nilai b: 5
masukkan nilai c: enam
input nya salah!
```

## SOAL 2

### Source Code:

```
try:

    bilangan = int(input("masukkan bilangan: "))

    nilai_bilangan = "positif" if bilangan > 0 else "negatif" if
bilangan < 0 else "nol"

    print(nilai_bilangan)

except:

    print("input salah!")
```

### Langkah-langkah :

1. Soal 4.2 diminta untuk menentukan bilangan yang kita input apakah termasuk bilangan positif, negatif, atau nol.
2. Terlebih dahulu buat untuk kita input bilangan
3. Lalu kita diminta untuk menggunakan ternary operator, menulis code lebih singkat dari yang biasanya
4. Jika bilangan > 0 maka akan mengeluarkan output positif, jika bilangan <0 akan mengeluarkan negatif, dan jika bilangan = 0 akan mengeluarkan nol.
5. Kita menggunakan **try – except** untuk menangani kesalahan input dari pengguna
6. Jika pengguna salah memasukkan input maka akan keluar input salah!

## Output :

```
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\tempCodeRunnerFile.py"
masukkan bilangan: 30
positif
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\tempCodeRunnerFile.py"
masukkan bilangan: -3
negatif
PS C:\pra alpro\minggu04> 0
0
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\tempCodeRunnerFile.py"
masukkan bilangan: lima
input salah!
PS C:\pra alpro\minggu04> █
```

## SOAL 3

### Source code: (menggunakan list)

try :

```
    bulan = int(input("Masukkan bulan: "))
```

```
    if bulan == 2:
```

```
        print('Jumlah hari 29')
```

```
    elif bulan in [1, 3, 5, 7, 8, 10, 12]:
```

```
        print('Jumlah hari 31')
```

```
    elif bulan in [4, 6, 9, 11]:
```

```
        print('Jumlah hari 30')
```

except:

```
    print('input yang anda masukkan salah!')
```

### Langkah-langkah:

1. Kita disuru untuk menghitung jumlah hari saat kita input bulan ke berapa pada tahun 2020.
2. Langkah pertama yang kita buat adalah membuat variabel untuk input bulan yang inginkan.
3. Kita menggunakan if elif karena memiliki kondisi lebih dari 1.
4. untuk bulan febuari memiliki jumlah hari yang berbeda, jadi kita membuat if bulan 2 terlebih dahulu.
5. Kita menggunakan list agar memudahkan untuk menyimpan item yang ada. List dapat digunakan dengan tanda kurung siku[].
6. Pada bulan 1, 3, 5, 7, 8, 10, 12 memiliki jumlah hari 31 sehingga kita dapat menyimpan item di dalam tanda kurung siku [1, 3, 5, 7, 8, 10, 12]
7. Bulan 4,6,9,11 memiliki jumlah hari 30 kita juga menyimpan item di dalam tanda kurung siku.
8. Pada soal ini kita diminta untuk menangani kesalahan input dari pengguna dengan cara menggunakan **try-except**, sehingga jika pengguna salah input maka output akan keluar "input yang ada masukkan salah!".

### Output:

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal03.py"
Masukkan bulan: 2
Jumlah hari 29
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal03.py"
Masukkan bulan: 3
Jumlah hari 31
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal03.py"
Masukkan bulan: 4
Jumlah hari 30
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\tempCodeRunnerFile.py"
Masukkan bulan: lima
input yang anda masukkan salah!
PS C:\pra alpro\minggu04> █
```

## **Cara ke-dua untuk menentukan jumlah hari:**

### **Source code:**

```
try:

    bulan = int(input("Masukkan bulan ke berapa: "))

    if bulan == 1:

        print("jumlah hari 31")

    elif bulan == 2:

        print("jumlah hari 29")

    elif bulan == 3:

        print("jumlah hari 31")

    elif bulan == 4:

        print("jumlah hari 30")

    elif bulan == 5:

        print("jumlah hari 31")

    elif bulan == 6:

        print("jumlah hari 30")

    elif bulan == 7:

        print("jumlah hari 31")

    elif bulan == 8:
```

```
        print("jumlah hari 31")
elif bulan == 9:
    print("jumlah hari 30")
elif bulan == 10:
    print("jumlah hari 31")
elif bulan == 11:
    print("jumlah hari 30")
elif bulan == 12:
    print("jumlah hari 31")
except:
    print("input yang anda masukkan salah!")
```

#### **Langkah-langkah:**

1. Disini kita menggunakan langkah manual, sehingga source code lebih panjang.
2. Kita membuat variabel input terlebih dahulu, agar pengguna dapat menginput bulan sesuai yang diinginkan.
3. Tetap sama, kita menggunakan if elif karena memiliki kondisi lebih dari 2.
4. Lalu kita melakukan percabangan if-elif,satu-persatu tiap bulannya.
5. Untuk menangani kesalahan input kita menggunakan **try-except** sehingga ketika pengguna salah memasukkan input maka output akan muncul "input yang anda masukkan salah!".



## Output:

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal03_2.py"
Masukkan bulan ke berapa: 4
jumlah hari 30
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal03_2.py"
Masukkan bulan ke berapa: 5
jumlah hari 31
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal03_2.py"
Masukkan bulan ke berapa: enam
input yang anda masukkan salah!
PS C:\pra alpro\minggu04> 
```

## SOAL 4

### Source code:

try:

```
a = int(input("Masukkan sisi 1: "))
b = int(input("Masukkan sisi 2: "))
c = int(input("Masukkan sisi 3: "))
```

```
if a == b == c:
```

```
    print("3 sisi sama")
```

```
elif a == b or a == c or b == c:
```

```
    print("2 sisi sama")
```

```
else:
```

```
    print("tidak ada yang sama")
```

except:

```
    print("input yang dimasukkan salah!")
```

### Langkah-langkah:

1. Pada soal ini, kita diminta untuk menentukan panjang ketiga sisi yang diinput oleh pengguna apakah ketiganya sama, atau 2 sisi yang sama atau tidak ada yang sama.
2. Langkah pertama, kita membuat 3 variabel untuk para pengguna input angka tiap sisi nya.
3. Langkah kedua, kita menggunakan percabangan if-elif-else. Jika a sama dengan b dan sama dengan c maka output yang dikeluarkan adalah ketiga sisi sama.
4. Langkah ketiga, kita menentukan untuk 2 sisi yang sama. Jika  $a=b$  atau  $a=c$  atau  $b=c$  akan mengeluarkan output dua sisi yang sama.
5. Langkah keempat, selain 3 atau 2 sisi yang sama maka tidak ada yang sama.
6. Disini kita juga diminta menggunakan **try-except** untuk menangani kesalahan input dari pengguna, jika pengguna salah memasukkan input maka muncul input yang dimasukkan salah.

### Output:

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal04.py"
Masukkan sisi 1: 20
Masukkan sisi 2: 20
Masukkan sisi 3: 10
2 sisi sama
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal04.py"
Masukkan sisi 1: 20
Masukkan sisi 2: 20
Masukkan sisi 3: 20
3 sisi sama
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal04.py"
Masukkan sisi 1: 30
Masukkan sisi 2: 20
Masukkan sisi 3: 10
tidak ada yang sama
```

```
PS C:\pra alpro\minggu04> python -u "c:\pra alpro\minggu04\soal04.py"
Masukkan sisi 1: enam
input yang dimasukkan salah!
```

