



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2024/2025

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	71220829
Nama Lengkap	TIMOTIUS EKANA TEJO KESUMA
Minggu ke / Materi	04 / Struktur Kontrol Percabangan

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2025

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.

MATERI 4.3.1 Boolean Expression dan Logical Operator

Boolean adalah bentuk yang menghasilkan keluaran True atau False. Boolean biasa ditentukan oleh beberapa operator perbandingan.

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

Boolean juga dapat digabungkan dengan logical operator seperti and, or, dan not.

MATERI 1.3.2 Bentuk-bentuk Percabangan

Bentuk percabangan dalam python terdapat 3 bentuk, yaitu: conditional, alternative dan chained conditional.

Conditional (if, jika):

if <kondisi>:

 <lakukan ini>

 <lakukan ini>

Alternative (if else, jika maka):

if <kondisi>:

<lakukan ini>

<lakukan ini>

...

else:

<lakukan itu>

<lakukan itu>

...

Chained Conditional (if else elif (else if), jika maka):

if <kondisi 1>:

<lakukan A1>

<lakukan A2>

...

elif <kondisi 2>

<lakukan B1>

<lakukan B2>

...

else

<lakukan ...>

<lakukan ...>

...

MATERI 1.3.3 Penanganan Kesalahan Input Menggunakan Exception Handling

Try and except digunakan untuk menangani inputan pengguna yang tidak sesuai dengan inputan yang dibutuhkan. Seperti: dibutuhkan inputan berupa integer, namun pengguna menginputkan string, maka code akan mengarahkan ke except.

Contoh:

```
1 try:
2     a = int(input("Masukkan angka: "))
3     c = a + 2
4     print(c)
5 except:
6     print("inputkan angka")
```

Input:

```
Masukkan angka: aa
```

Output:

```
inputkan angka
```

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.

SOAL 1

Implementasikan penanganan kesalahan input pengguna dari program-program pada Contoh 3.1, 3.2 dan 3.3.

4.1

```
1 try:
2     suhu = int(input("Masukkan suhu tubuh: "))
3     if suhu >= 38:
4         print("Anda demam")
5     else:
6         print("Anda tidak demam")
7 except:
8     print("Input tidak valid")
```

Output:

```
Masukkan suhu tubuh: a
Input tidak valid
```

4.2

```
1 try:
2     bilangan = int(input("Masukkan suatu bilangan: "))
3     if bilangan > 0:
4         print("Positif")
5     elif bilangan < 0:
6         print("Negatif")
7     elif bilangan == 0:
8         print("Nol")
9 except:
10    print("Input tidak valid")
```

Output:

```
Masukkan suatu bilangan: asdasd
Input tidak valid
```

4.3

```

1  try:
2      a = int(input("Masukkan bilangan pertama: "))
3      b = int(input("Masukkan bilangan kedua: "))
4      c = int(input("Masukkan bilangan ketiga: "))
5      if a > b and a > c:
6          print("Terbesar: ", a)
7      elif b > a and b > c:
8          print("Terbesar: ", b)
9      elif c > a and c > b:
10         print("Terbesar: ", c)
11 except:
12     print("Input tidak valid")

```

Output:

```

Masukkan bilangan pertama: asd
Input tidak valid

```

SOAL 2

Implementasikan percabangan pada Contoh 3.2 (Positif-Negatif) menggunakan ternary operator.

```

1  bilangan = int(input("Masukkan suatu bilangan: "))
2  print("Positif" if bilangan > 0 else "Negatif" if bilangan < 0 else "Nol")

```

Output:

```

Masukkan suatu bilangan: 10
Positif

```

SOAL 3

```

1  try:
2      bulan = int(input("Masukkan bulan: "))
3      if bulan in [1, 3, 5, 7, 8, 10, 12]:
4          print("Jumlah hari :", 31)
5      elif bulan in [4, 6, 9, 11]:
6          print("Jumlah hari :", 30)
7      elif bulan == 2:
8          print("Jumlah hari :", 29)
9  except:
10     print("bulan yang diinputkan oleh pengguna tersebut tidak valid")

```

Output:

```

Masukkan bulan: asdas
bulan yang diinputkan oleh pengguna tersebut tidak valid

```

```

Masukkan bulan: 2
Jumlah hari : 29

```

SOAL 4

```
1  try:
2      sisi1 = int(input("Masukkan sisi 1: "))
3      sisi2 = int(input("Masukkan sisi 2: "))
4      sisi3 = int(input("Masukkan sisi 3: "))
5
6      if sisi1 == sisi2 == sisi3:
7          print("3 sama sisi")
8      elif sisi1 == sisi2 or sisi2 == sisi3 or sisi1 == sisi3:
9          print("2 sama sisi")
10     else:
11         print("Tidak ada yang sama")
12 except:
13     print("input tidak valid")
```

Output:

```
Masukkan sisi 1: 22
Masukkan sisi 2: 22
Masukkan sisi 3: 3
2 sama sisi
```

Link GitHub:

https://github.com/TimotiusEkanaT/Alpro/tree/main/Laporan_4