



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71210695
Nama Lengkap	Cahaya Sampebua
Minggu ke / Materi	03 / Struktur Kontrol Percabangan

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.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2024

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 merupakan ekspresi yang menghasilkan nilai boolean TRUE dan FALSE. Boolean expression disusun menggunakan operator-operator perbandingan yaitu ==, !=, >, <, >=, <=, is, is not. Apa yang perlu diperhatikan pada Boolean expression adalah variable yang dibandingkan menggunakan operator pembandingan yang sesuai dengan permasalahan agar diperoleh Boolean expression yang diharapkan (TRUE atau FALSE).

Contoh:

Untuk mendapatkan nilai A maka nilai hasil ujian >= 85

```
>>> nilai_Ani = 80
>>> nilai_Ani >= 85
False
>>>
```

Contoh:

Membandingkan dua nilai apakah sama atau tidak sama menggunakan is

```
>>> Berat_Roni = 25
>>> Berat_Joni = 25
>>> Berat_Roni is Berat_Joni
True
>>>
```

Contoh:

Syarat orang yang boleh mendaftar polisi adalah umur >= 17 tahun dan tinggi badan untuk laki-laki >= 160 cm

```
>>> umur = 18
>>> tinggi_badan = 162
>>> umur >= 17 and tinggi_badan >= 160
True
>>>
```

Bentuk-bentuk Percabangan

Bentuk percabangan pada Python menggunakan fungsi if. Pengguna perlu memasukkan kondisi dalam bentuk Boolean Expression, jika bernilai True, instruksi akan dilakukan, jika False, instruksi tidak akan dilakukan.

Ada 3 jenis bentuk percabangan:

- Conditional

Conditional berarti percabangan hanya mempunyai satu kondisi. Bila kondisi tidak terpenuhi, maka Python akan men-skip perintah if dan akan langsung menuju ke instruksi selanjutnya.

```
Type "help", "copyright", or
on.
>>> x = 11
>>> if x > 10:
...     x = x*2
...
>>> x
22
>>> █
```

Figure 1 conditional

- Alternative

Alternative artinya ada 2 kondisi: kondisi yang pertama dan kondisi bila kondisi yang pertama gagal. Terdiri dari if dan else. Bila kondisi pertama tidak terpenuhi, Python akan langsung men-skip perintah if dan langsung mengeksekusikan perintah else.

```
on.
>>> x = 11
>>> if x > 12:
...     x = x*2
... else:
...     x = x + 10
...
>>> x
21
>>> █
```

Figure 2 alternative

- Chained conditional

Chained conditional artinya ada lebih dari 2 kondisi (if-elif-else [optional]). Jika kondisi pertama tidak terpenuhi, maka akan di-skip ke kondisi ke-2 dan seterusnya. Jika kondisi pada perintah if dan elif tidak terpenuhi, Python akan men-skip perintah if dan elif dan langsung ke instruksi selanjutnya. Jika memakai else, Python akan mengeksekusikan perintah else.

Contoh:

```

on.
>>> x = 11
>>> if x > 11:
>>>     x = x*2
>>> elif x == 11:
>>>     x = x*3>>> ... ...
>>>
>>> x
33
>>> █

```

Figure 3 chained conditional tanpa menggunakan else

```

on.
>>> x = 10
>>> if x > 11:
>>>     x = x*2
>>> elif x == 11:
>>>     x = x*3
>>> else:
>>>     x = x+10
>>>
>>> x
20
>>> █

```

Figure 4 chained conditional dengan menggunakan else

Selain dari 3 jenis percabangan tersebut, ada juga bentuk ternary yang hanya memiliki 1 baris:

```

Type "help", "copyright", "credits" or "license()"
on.
>>> x = 10
>>> x = x*2 if x > 11 else x + 11
>>> x
21
>>> █

```

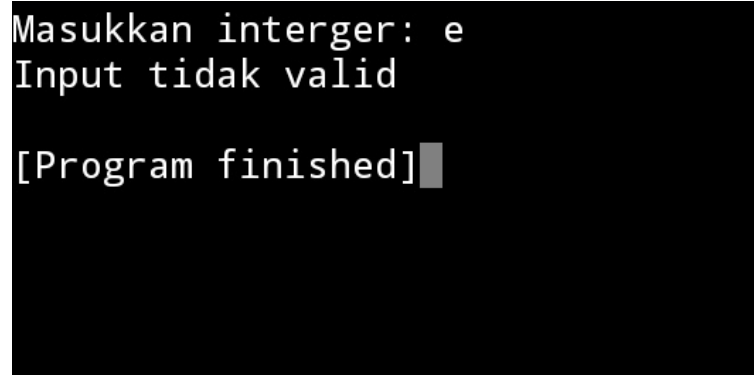
Figure 5 ternary operator

Penanganan Kesalahan Input Menggunakan Exception Handling

Misalkan, pengguna memasukkan input yang salah: pengguna diminta memasukkan input berupa interger, tetapi malah memasukkan huruf karena suatu alasan tertentu (*typo*, misalnya), maka akan membuat program kita menjadi error. Idealnya, kita tidak mau program kita error saat dipakai oleh pengguna. Oleh karena itu, untuk menangani hal tersebut dapat menggunakan fungsi try – except. Python akan mencoba untuk mengeksekusikan perintah try. Apabila terjadi error (misalnya error input pengguna), Python langsung akan berhenti dan langsung mengeksekusikan perintah except.

```
try:
    a = int(input("Masukkan interger: "))
    print(a)
except:
    print("Input tidak valid")
```

Figure 6 try except



```
Masukkan interger: e
Input tidak valid

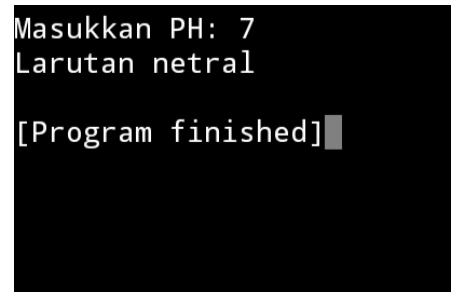
[Program finished]
```

Figure 7 output apabila pengguna memasukkan input yang salah

Contoh Masalah-masalah Percabangan

Menentukan larutan asam, basa, atau netral

```
try:
    ph = float(input("Masukkan PH: "))
    if ph == 7:
        print("Larutan netral")
    elif ph > 7:
        print("Larutan basa")
    elif ph < 7 and ph >= 0:
        print("Larutan asam")
    else:
        print("Invalid input")
except:
    print("Invalid input")
```



```
Masukkan PH: 7
Larutan netral

[Program finished]
```

Figure 9 output netral

Figure 8 source code

```
Masukkan PH: g
Invalid input

[Program finished]
```

Figure 8 output error

```
Masukkan PH: -1
Invalid input

[Program finished]
```

Figure 11 output error 2

```
Masukkan PH: 3
Larutan asam
```

Figure 12 output asam

```
Masukkan PH: 9
Larutan basa
```

Figure 3 output basa

Pertama menginputkan ph larutan. Kemudian menentukan kondisi: jika ph kurang dari 7 dan tidak kurang dari 0, larutan asam. Jika ph 7, larutan netral. Jika ph lebih dari 7, larutan basa. Selain itu maka input salah. Untuk mengatasi error pada inputan pengguna, pakai try except. Jika pengguna salah memasukkan input, akan keluar pemberitahuan bahwa input yang dimasukkan tidak valid.

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/cahaya-sampebua/Latihan-Mandiri-Pertemuan-3-ParakAIPro.git>

SOAL 1

- 3.1

Pertama, kita inputkan dulu suhu. Setelah itu, tuliskan kondisi: jika suhu 38 ke atas maka demam, jika tidak maka tidak demam. Jika semisal user salah memasukkan input (semisalnya huruf), Python akan langsung berhenti mengeksekusikan perintah dan akan langsung menampilkan peringatan bahwa input data yang dimasukkan salah.

```
suhuInputan = input("Masukkan suhu tubuh: ")
try:
    suhu = int(suhuInputan)
    if suhu >= 38:
        print("Anda demam")
    else:
        print("Anda tidak demam")
except:
    print("Maaf, Inputan Yang Anda Masukkan Salah")
```

```
Masukkan suhu tubuh: 36
Anda tidak demam
```

Output jika pengguna menginput data suhu yang sesuai

```
Masukkan suhu tubuh: sdfes45
Maaf, Inputan Yang Anda Masukkan Salah
```

Output jika pengguna menginput data suhu yang tidak sesuai

- 3.2

Pertama, masukkan input bilangan. Kemudian, tuliskan kondisi yang dikehendaki: bila bilangan adalah 0, tampilkan *Nol*; bila bilangan lebih dari 0, tampilkan *Positif*; bila bilangan kurang dari 0, tampilkan *negative*. Untuk menangani error input, pakai fungsi try-except. Bila terjadi error, akan dimunculkan pesan yang berisi bahwa data yang diinputkan tidak valid.

```

inputbilangan = input("Masukkan suatu bilangan: ")
try:
    bilangan = int(inputbilangan)
    if bilangan > 0:
        print("Positif")
    elif bilangan < 0:
        print("Negatif")
    elif bilangan == 0:
        print("Nol")
except:
    print("Input bilangan tidak sesuai")

```

Masukkan suatu bilangan: 10
Positif

Output jika bilangan > 0

Masukkan suatu bilangan: -1
Negatif

Output jika bilangan < 0

Masukkan suatu bilangan: 0
Nol

Output jika bilangan == 0

Masukkan suatu bilangan: sdf5435
Input bilangan tidak sesuai

Output jika pengguna menginput data suhu yang tidak sesuai

- 3.3
Pertama, masukkan input bilangan a, b, c. kemudian, tulis kondisinya: jika a lebih dari b dan c, a adalah bilangan terbesar. Jika b lebih besar dari a dan c, b adalah bilangan terbesar. Bila c lebih besar dari a dan b, c adalah bilangan terbesar. Dengan memakai fungsi try-except, jika terjadi kesalahan penginputan, akan muncul pesan yang berbunyi bahwa input yang dimasukkan salah.


```

masukkan_a = input("Masukkan bilangan pertama: ")
masukkan_b = input("Masukkan bilangan kedua: ")
masukkan_c = input("Masukkan bilangan ketiga: ")

# secara berurutan tulis kriteria untuk a, b, dan c
try:
    a = int(masukkan_a)
    b = int(masukkan_b)
    c = int(masukkan_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 a, b, c tidak ada yang sesuai")

```

Masukkan bilangan pertama: 1
 Masukkan bilangan kedua: 123
 Masukkan bilangan ketiga: 10
 Terbesar: 123
 Output jika sesuai dengan inputan

Masukkan bilangan pertama: adsf345
 Masukkan bilangan kedua: 234e
 Masukkan bilangan ketiga: 45twrfds
 Input a, b, c, tidak ada yang sesuai

Output jika inputan a, b, c tidak sesuai dengan inputan

SOAL 2

Pertama inputkan bilangan. Kemudian tentukan kondisi: bila bilangan adalah 0, tampilkan *Nol*; bila bilangan lebih dari 0, tampilkan *Positif*; bila bilangan kurang dari 0, tampilkan *negatif*. Karena menggunakan Ternary Operator, maka fungsi if-elif-else ditulis seperti pada figure 20

```

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

```

Masukkan suatu bilangan: 13
 Positif

Output jika bilangan > 0

Masukkan suatu bilangan: -1
 Negatif

Output jika bilangan < 0

Masukkan suatu bilangan: 0
No1

Output jika bilangan == 0

SOAL 3

Pertama, inputkan bulan ke-berapa (1 – 12). Kemudian masukkan kondisi yang diinginkan:

- Jika bulan antara 1 – 7:
Kecuali Februari (29 hari), semua bulan yang genap memiliki 30 hari sehingga jika bulan modulus 2 hasilnya 0, bulan tersebut adalah bulan urutan genap dan pasti mempunyai 30 hari. Jika bulan modulus 2 hasilnya 1, bulan tersebut mempunyai urutan ganjil dan pasti mempunyai 31 hari.
- Jika bulan antara 8 – 12:
Semua bulan yang ganjil memiliki 30 hari sehingga jika bulan modulus 2 hasilnya 1, bulan tersebut adalah bulan urutan ganjil dan pasti mempunyai 30 hari. Jika bulan modulus 2 hasilnya 0, bulan tersebut mempunyai urutan genap dan pasti mempunyai 31 hari.

Dengan memakai fungsi try-except, jika ada kesalahan input maka akan ditampilkan pesan bahwa data yang diinputkan salah. Jika pengguna memasukkan angka interger yang tidak memenuhi kondisi di atas, maka akan dieksekusikan perintah else, yaitu 7/0. Hal ini disengaja supaya menghasilkan error sehingga perintah except dapat dijalankan.

```
try:
    bulan = int(input("Masukkan bulan (1-12): "))
    if bulan < 8 and bulan > 0:
        if bulan == 2:
            hari = 29
        elif bulan % 2 == 1:
            hari = 31
        elif bulan % 2 == 0:
            hari = 30
    elif bulan < 13 and bulan > 0:
        if bulan % 2 == 1:
            hari = 30
        else:
            hari = 31
    else:
        raise ValueError("Bulan yang diinputkan tidak valid")
    print("Jumlah hari:", hari)
except ValueError:
    print("Bulan yang diinputkan tidak valid")
```

Masukkan bulan (1-12): 7
Jumlah hari: 31

Output jika inputannya sesuai

```
Masukkan bulan (1-12): 13  
Bulan yang diinputkan tidak valid
```

Output jika inputannya tidak valid

SOAL 3

Pertama masukkan input untuk sisi 1, sisi 2, dan sisi 3. Kemudian, masukkan kondisi yang diinginkan: jika semua sisi sama, tampilkan *3 sisi sama*. Jika tidak ada sisi yang sama, tampilkan *Tidak ada yang sama*. Selain itu, tampilkan *2 sisi sama*. Jika pengguna salah menginputkan data, maka akan ditampilkan pesan bahwa data yang diinputkan tidak valid.

```
try:  
    sisi1 = int(input("Masukkan sisi 1: "))  
    sisi2 = int(input("Masukkan sisi 2: "))  
    sisi3 = int(input("Masukkan sisi 3: "))  
  
    if sisi1 != sisi2 and sisi2 != sisi3 and sisi3 != sisi1:  
        print("Tidak ada yang sama")  
  
    elif sisi1 == sisi2 and sisi2 == sisi3 and sisi1 == sisi3:  
        print("3 sisi sama")  
  
    elif sisi1 == sisi2 or sisi2 == sisi3 or sisi1 == sisi3:  
        print("2 sisi sama")  
except:  
    print("Inputan yang anda masukkan salah")
```

```
Masukkan sisi 1: 14  
Masukkan sisi 2: 18  
Masukkan sisi 3: 11  
Tidak ada yang sama
```

Output jika inputan tidak ada sisi yang sama

```

try:
    sisi1 = int(input("Masukkan sisi 1: "))
    sisi2 = int(input("Masukkan sisi 2: "))
    sisi3 = int(input("Masukkan sisi 3: "))

    if sisi1 != sisi2 and sisi2 != sisi3 and sisi3 != sisi1:
        print("Tidak ada yang sama")

    elif sisi1 == sisi2 and sisi2 == sisi3 and sisi1 == sisi3:
        print("3 sisi sama")

    elif sisi1 == sisi2 or sisi2 == sisi3 or sisi1 == sisi3:
        print("2 sisi sama")
except:
    print("Inputan yang anda masukkan salah")

```

```

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

```

Output jika inputan dengan 3 sisi yang sama

```

try:
    sisi1 = int(input("Masukkan sisi 1: "))
    sisi2 = int(input("Masukkan sisi 2: "))
    sisi3 = int(input("Masukkan sisi 3: "))

    if sisi1 != sisi2 and sisi2 != sisi3 and sisi3 != sisi1:
        print("Tidak ada yang sama")

    elif sisi1 == sisi2 and sisi2 == sisi3 and sisi1 == sisi3:
        print("3 sisi sama")

    elif sisi1 == sisi2 or sisi2 == sisi3 or sisi1 == sisi3:
        print("2 sisi sama")
except:
    print("Inputan yang anda masukkan salah")

```

```

Masukkan sisi 1: 8
Masukkan sisi 2: 9
Masukkan sisi 3: 8
2 sisi sama

```

Output jika inputan dengan 2 sisi yang sama

```
try:
    sisi1 = int(input("Masukkan sisi 1: "))
    sisi2 = int(input("Masukkan sisi 2: "))
    sisi3 = int(input("Masukkan sisi 3: "))

    if sisi1 != sisi2 and sisi2 != sisi3 and sisi3 != sisi1:
        print("Tidak ada yang sama")

    elif sisi1 == sisi2 and sisi2 == sisi3 and sisi1 == sisi3:
        print("3 sisi sama")

    elif sisi1 == sisi2 or sisi2 == sisi3 or sisi1 == sisi3:
        print("2 sisi sama")
except:
    print("Inputan yang anda masukkan salah")
```

Masukkan sisi 1: 12
Masukkan sisi 2: dasd
Inputan yang anda masukkan salah

Output jika inputan yang dimasukkan salah