



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71220895
Nama Lengkap	Drestanta Dipta Jalu Prakasya
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.

MATERI 1

Dalam Python, kita dapat membuat pernyataan perbandingan dengan menggunakan operator perbandingan seperti `==`, `!=`, `>`, `<`, `>=`, dan `<=`. Ekspresi boolean adalah jenis ekspresi yang menghasilkan nilai Benar atau Palsu. Berikut contohnya:

- Lulus dengan IPK minimum 2.25:
`IPK >= 2.25`
- Golden Button untuk Youtuber dengan subscriber lebih dari 1 juta:
`Subscriber > 1000000`
- Tilang untuk pengendara berkecepatan lebih dari 90 km/jam:
`Kecepatan > 90`
- Rollercoaster hanya untuk yang tinggi lebih dari 110:
`Tinggi > 110`
- Nilai ujian Hanna dan Robby sama:
`Hanna == Robby`
- Sepatu Junaedi ditambah Ricky lebih banyak dari pada Arnold:
`Junaedi+Ricky > Arnold`
- Rollercoaster untuk penumpang dengan usia minimal 10 tahun dan tinggi minimal 110 cm:
`Usia >= 10 and tinggi >= 110`
- Diskon diberikan kepada member atau pembeli lebih dari Rp. 500.000:
`Member or pembelian > 500000`

Kita dapat menyederhanakan pengecekan kondisi dalam program Python dengan menggunakan ekspresi boolean dan operator logis.

MATERI 2

Tiga jenis percabangan umum yang tersedia dalam Python: conditional, alternative, dan chained conditional.

1. Conditional:

If <kondisi>:
 <lakukan ini>
 <lakukan ini>

Contoh:

```
if nilai > 70:  
    print("Selamat anda lulus!!!")
```

2. Alternative:

If <kondisi>:
 <lakukan ini>
 <lakukan ini>
Else:
 <lakukan ini>
 <lakukan ini>

Contoh:

```
if nilai > 60:  
    print("Lulus")  
else:  
    print("Tidak Lulus")
```

3. Chained Conditional:

if <kondisi 1>:
 <lakukan A1>
 <lakukan A2>
elif <kondisi 2>:
 <lakukan B1>
 <lakukan B2>
elif <kondisi 3>:
 <lakukan C1>
 <lakukan C2>
else:
 <lakukan ...>
 <lakukan ...>

Contoh:

```
if pembelian > 1000000:  
    diskon = 0.3  
elif pembelian > 500000 and pembelian <= 1000000:  
    diskon = 0.2  
elif pembelian >= 10000 and pembelian <= 500000:  
    diskon = 0.15  
else:  
    diskon = 0
```

MATERI 3

Program mencoba mengonversi input pengguna menjadi integer, menentukan kategori usia sesuai aturan yang ditetapkan, jika berhasil. Namun, jika prosesnya gagal, seperti ketika pengguna memasukkan teks yang tidak dapat diubah menjadi bilangan bulat, program akan melompat ke blok except dan menampilkan pesan kesalahan.

```
inputuser = input("Masukkan usia anda: ")
try:
    usia = int(inputuser)
    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("Anda salah memasukkan input usia")
```

Program dapat menangani input yang tidak sesuai dengan cara yang lebih halus tanpa mengalami kegagalan total dengan blok try-except. Untuk meningkatkan pengalaman pengguna dan membantu program tetap berjalan bahkan jika terjadi kesalahan input, pesan kesalahan akan ditampilkan dan program tidak akan berhenti atau menghasilkan traceback.

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/Drestanta/Laprak-3.git>

SOAL 1.1

Kita dapat menggunakan blok try-except untuk menangani kesalahan input pengguna pada program ini. Ini akan memeriksa apakah input adalah bilangan bulat atau tidak, dan jika input tidak sesuai, kita akan menangkap kesalahan dan memberikan pesan kesalahan kepada pengguna.

Ini adalah contoh pelaksanaannya:

```
In [8]: try:
        suhu = float(input("Masukkan suhu tubuh: "))
        if suhu >= 38:
            print("Anda demam")
        elif suhu <= 35:
            print("Anda hipotermia")
        else:
            print("Anda tidak demam")
    except:
        print("Input tidak valid.")

Masukkan suhu tubuh: demam
Input tidak valid.
```

Program ini juga dapat menerima input dalam bentuk bilangan desimal dengan float(input(...)). Jika input tidak dapat diubah menjadi bilangan, seperti ketika pengguna memasukkan teks, maka blok except akan menangkap kesalahan dan memberikan pesan kesalahan yang sesuai.

SOAL 1.2

Contoh implementasi:

```
In [3]: try:
        angka = int(input("Masukkan suatu bilangan: "))

        if angka > 0:
            print("Positif")
        elif angka < 0:
            print("Negatif")
        else:
            print("Nol")

    except:
        print("Error: Masukkan harus berupa bilangan bulat")
```

```
Masukkan suatu bilangan: -90
Negatif
```

Kesalahan yang mungkin terjadi jika pengguna memasukkan input yang bukan bilangan bulat diatasi dengan penambahan try dan except. Jika terjadi kesalahan, program akan mencetak pesan kesalahan yang sesuai, dan jika input sesuai, program akan mengevaluasi apakah bilangan tersebut positif, negatif, atau nol sesuai dengan kriteria yang diberikan.

SOAL 1.3

Kita dapat menggunakan blok try dan except untuk menangkap kesalahan yang mungkin terjadi saat pengguna memasukkan input yang tidak valid dalam program ini. Ini adalah contoh implementasinya:

```
In [2]: try:
    bilangan1 = int(input("Masukkan bilangan pertama: "))
    bilangan2 = int(input("Masukkan bilangan kedua: "))
    bilangan3 = int(input("Masukkan bilangan ketiga: "))

    if bilangan1 > bilangan2 and bilangan1 > bilangan3:
        print("Terbesar: ", bilangan1)
    elif bilangan2 > bilangan1 and bilangan2 > bilangan3:
        print("Terbesar: ", bilangan2)
    elif bilangan3 > bilangan1 and bilangan3 > bilangan2:
        print("Terbesar: ", bilangan3)
    elif bilangan1 == bilangan2 == bilangan3:
        print("Semua sama besar: ", bilangan1)

except:
    print("Input tidak valid.")

Masukkan bilangan pertama: 11
Masukkan bilangan kedua: 22
Masukkan bilangan ketiga: 11
Terbesar: 22
```

Dengan menambahkan blok try dan except, program akan menangkap kesalahan jika pengguna memasukkan input yang bukan angka bulat. Jika konversi ke integer tidak berhasil, kesalahan akan ditangkap. Kita juga menangkap kesalahan umum untuk menangani kesalahan lainnya yang mungkin terjadi.

SOAL 2

Berikut percabangan diimplementasikan menggunakan operator ternary:

```
In [1]: angka = int(input("Masukkan suatu bilangan: "))
        nilai = "Positif" if angka > 0 else "Negatif" if angka < 0 else "Nol"
        print(nilai)

Masukkan suatu bilangan: 88
Positif
```

Dalam kode di atas, "nilai" akan ditunjukkan sebagai "Positif" jika bilangan lebih dari 0, "Negatif" jika bilangan kurang dari 0, dan "Nol" jika bilangan sama dengan 0. Setelah itu, nilai hasil akan dicetak.

SOAL 3

Berikut codenya:

```
In [1]: def jumlah_hari(bulan):
        if bulan == 2:
            return 28
        elif bulan in [4, 6, 9, 11]:
            return 30
        elif bulan in [1, 3, 5, 7, 8, 10, 12]:
            return 31
        else:
            return None

        try:
            input_bulan = int(input("Masukkan nomor bulan (1-12): "))

            if 1 <= input_bulan <= 12:
                hasil = jumlah_hari(input_bulan)
                if hasil is not None:
                    print(f"Jumlah hari: {hasil}")
                else:
                    print("Bulan tidak valid.")
            else:
                print("Nomor bulan tidak valid.")
        except:
            print("Input yang anda masukkan salah")
```

```
Masukkan nomor bulan (1-12): 12
Jumlah hari: 31
```

Fungsi jumlah_hari digunakan oleh program di atas untuk mengembalikan jumlah hari berdasarkan nomor bulan yang dimasukkan. Program juga menangani kesalahan jika pengguna memasukkan angka atau nomor bulan yang salah.

SOAL 4

Program Python yang meminta pengguna untuk memasukkan panjang ketiga sisi segitiga dan menampilkan pesan yang sesuai:

```
In [2]: try:
        sisi1 = float(input("Masukkan panjang sisi 1: "))
        sisi2 = float(input("Masukkan panjang sisi 2: "))
        sisi3 = float(input("Masukkan panjang sisi 3: "))

        if sisi1 == sisi2 == sisi3:
            print("3 sisi sama")
        elif sisi1 == sisi2 or sisi1 == sisi3 or sisi2 == sisi3:
            print("2 sisi sama")
        else:
            print("Tidak ada yang sama")
    except:
        print("Input tidak valid.")

Masukkan panjang sisi 1: 11
Masukkan panjang sisi 2: 11
Masukkan panjang sisi 3: 11
3 sisi sama
```

Untuk memastikan bahwa pengguna memasukkan input yang valid, program di atas menampilkan pesan kesalahan jika mereka memasukkan angka.