


<p>Nama: Aini Rihhadatul Aisy</p> <p>NIM: 064102400024</p>	 <p>Praktikum Algoritma & Pemrograman</p>	<p>MODUL 5</p> <p>Nama Dosen: Binti solihah, S.T, M.KOM</p>
<p>Hari/Tanggal: Jum'at, 18 Oktober 2024</p>		<p>Nama Asisten Labratorium:</p> <ol style="list-style-type: none"> 1. Yustianas Rombon - 064002300015 2. Vira Aditya Kurniawan - 065002300012

Latihan – Struktur Kendali

1. Teori Singkat

Operasi Perulangan

Perulangan dalam bahasa pemrograman berfungsi menugaskan komputer untuk melakukan sesuatu secara berulang-ulang. Terdapat dua jenis perulangan dalam bahasa pemrograman python, yaitu perulangan dengan *while* dan *for*. Perulangan *for* disebut counted loop (perulangan yang terhitung), sementara perulangan *while* disebut uncounted loop (perulangan yang tak terhitung). Perbedaannya adalah perulangan *for* biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya. Sementara *while* untuk perulangan yang memiliki syarat dan tidak tentu berapa banyak perulangannya. Contohnya:

Berikut contoh program yang menggunakan perulangan *while*:

Source Code

```
ulangan = 0

while (ulangan < 10):
    print("Kuulangan ", ulangi, " kali")
    ulangi = ulangi + 1
```

Output





Berikut contoh program yang menggunakan perulangan *for*:

Source Code



```

for i in range(5):
    print("Nilai i adalah ", i)

for i in range(3,10):
    print("Kuulangi sebanyak ", i,
"kali")

for i in range(30,20,-2):
    print("Aku mundur dari 30 ", i)

```

Output

```

Nilai i adalah 0
Nilai i adalah 1
Nilai i adalah 2
Nilai i adalah 3
Nilai i adalah 4
Kuulangi sebanyak 3 kali
Kuulangi sebanyak 4 kali
Kuulangi sebanyak 5 kali
Kuulangi sebanyak 6 kali
Kuulangi sebanyak 7 kali
Kuulangi sebanyak 8 kali
Kuulangi sebanyak 9 kali
Aku mundur dari 30 30
Aku mundur dari 30 28
Aku mundur dari 30 26
Aku mundur dari 30 24
Aku mundur dari 30 22

```

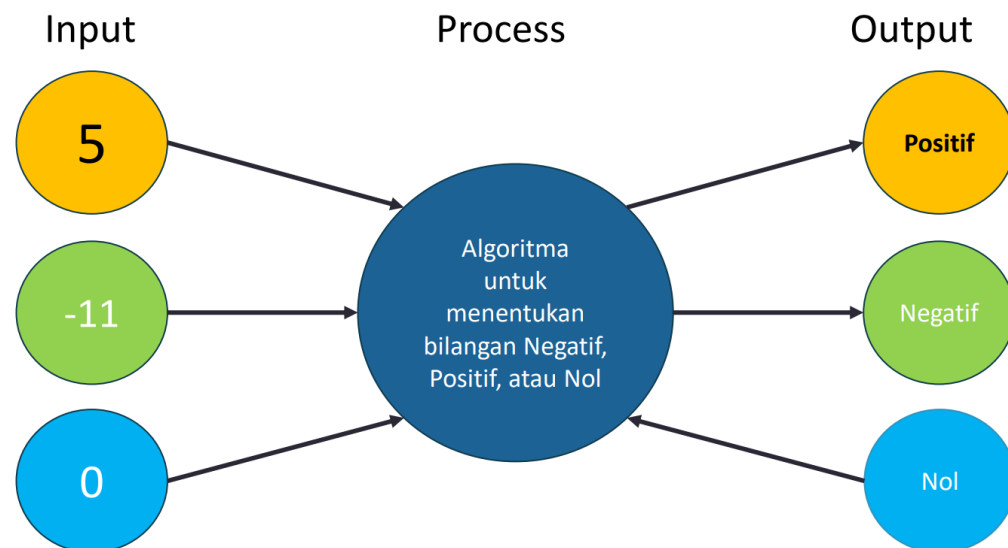


IPO (Input Process Output)

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)



Pseudocode

Pseudocode adalah suatu bentuk deskripsi informal yang mirip dengan bahasa manusia dan digunakan untuk menggambarkan algoritma atau proses secara naratif. Ini tidak terikat pada bahasa pemrograman tertentu, tetapi memberikan panduan tentang langkah-langkah yang harus diambil dalam suatu algoritma dengan bahasa yang lebih mudah dimengerti.



Contoh PseudoCode

Inisiasi Variabel:

```
N      = 0
total  = 0.0
```

Pengulangan:

```
UNTUK i DARI 1 SAMPAI 10 LANGKAH 2
  CETAK i
END UNTUK
```

Pengkondisional (Conditional):

```
JIKA nilai > 10
  CETAK "Nilai lebih dari 10"
SELAINNYA JIKA nilai = 10
  CETAK "Nilai sama dengan 10"
SELAINNYA
  CETAK "Nilai kurang dari 10"
AKHIR JIKA
```

Fungsi atau Prosedur:

```
FUNGSI tambah(a, b)
  KEMBALIKAN a + b
AKHIR FUNGSI
```

Contoh Lengkap:

```
DEKLARASI variabel n, bilangan, total, rata_rata FLOAT
MINTA "Masukkan jumlah bilangan: " SIMPAN
total = 0.0

UNTUK i DARI 1 SAMPAI n
  MINTA "Masukkan bilangan ke-" + i + ": " SIMPAN bilangan
  total = total + bilangan
END UNTUK

rata_rata = total / n
CETAK "Rata-rata adalah: " + rata_rata
```

2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)

3. Elemen Kompetensi

a. Latihan pertama

Buatlah program fungsi untuk merata-ratakan nilai sesuai dengan kategori huruf yang diinputkan dimana aturannya adalah sebagai berikut: (Exercise 66)

A = 4.00

A- = 3.75

B+ = 3.50

B = 3.00

B- = 2.75

C+ = 2.50

C = 2.00

C- = 1.75

D = 1.50

E = 1.25

Implementasikan apa yang kalian pelajari pada materi-materi di modul-modul sebelumnya



Input Process Output

Input:

1. Jumlah mapel: Jumlah mata pelajaran yang dinilai.
2. Grade: Nilai untuk setiap mata pelajaran (A, A-, B+, B, B-, C+, C, C-, D, E).

Proses:

1. Inisialisasi total_nilai dan counter.
2. Gunakan perulangan untuk menerima input grade hingga jumlah mapel tercapai:
 - o Tentukan nilai numerik berdasarkan grade.
 - o Tambahkan nilai ke total_nilai dan tingkatkan counter.
3. Hitung rata-rata nilai dengan membagi total_nilai dengan jumlah_mapel.

Output:

1. Tampilkan rata-rata nilai.

Pseudocode

1. Minta user memasukkan jumlah_mapel
2. Set total_nilai <- 0
Set counter <- 0
3. Selama counter < jumlah_mapel:
 - a. Minta user memasukkan inputNilai (grade huruf)
 - Jika inputNilai == "A", tambahkan 4.00 ke total_nilai, tambah counter
 - Jika inputNilai == "A-", tambahkan 3.75 ke total_nilai, tambah counter
 - Jika inputNilai == "B+", tambahkan 3.50 ke total_nilai, tambah counter
 - Jika inputNilai == "B", tambahkan 3.00 ke total_nilai, tambah counter
 - Jika inputNilai == "B-", tambahkan 2.75 ke total_nilai, tambah counter
 - Jika inputNilai == "C+", tambahkan 2.50 ke total_nilai, tambah counter
 - Jika inputNilai == "C", tambahkan 2.00 ke total_nilai, tambah counter
 - Jika inputNilai == "C-", tambahkan 1.75 ke total_nilai, tambah counter
 - Jika inputNilai == "D", tambahkan 1.50 ke total_nilai, tambah counter
 - Jika inputNilai == "E", tambahkan 1.25 ke total_nilai, tambah counter
4. Hitung rata_rata dengan cara total_nilai DIBAGI jumlah_mapel
5. Cetak rata-rata nilai



Source Code

```
+ Code + Text

[7] jumlah_mapel = int(input("Masukkan jumlah mapel: "))
total_nilai = 0
counter = 0
while counter < jumlah_mapel:
    inputNilai = input("Masukkan Grade: ")

    if inputNilai == "A":
        counter += 1
        total_nilai += 4.00
        continue
    elif inputNilai == "A-":
        counter += 1
        total_nilai += 3.75
        continue
    elif inputNilai == "B+":
        counter += 1
        total_nilai += 3.50
        continue
    elif inputNilai == "B":
        counter += 1
        total_nilai += 3.00
        continue

    elif inputNilai == "B-":
        counter += 1
        total_nilai += 2.75
        continue
    elif inputNilai == "C+":
        counter += 1
        total_nilai += 2.50
        continue
    elif inputNilai == "C":
        counter += 1
        total_nilai += 2.0
        continue
    elif inputNilai == "C-":
        counter += 1
        total_nilai += 1.75
        continue
    elif inputNilai == "D":
        counter += 1
        total_nilai += 1.50
        continue
    elif inputNilai == "E":
        counter += 1
```




```
elif inputNilai == "E":
    counter += 1
    total_nilai += 1.25
    continue

rata_rata = total_nilai / jumlah_mapel
print("Rata-rata nilai:", rata_rata)
```

Output

```
➡ Masukkan jumlah mapel: 3
Masukkan Grade: A
Masukkan Grade: B
Masukkan Grade: C
Rata-rata nilai: 3.0
```

b. Latihan Kedua

Buatlah program untuk menghitung Tiket Masuk Kebun Binatang Berdasarkan Umur beserta Pembayaranannya dengan aturan sebagai berikut: (Exercise 67)

- Umur yang kurang dari atau sama dengan 2 tahun digratiskan
- Umur yang lebih dari atau sama dengan 3 tahun hingga umur yang kurang dari atau sama dengan 12 tahun seharga 14 dollar
- Umur lebih dari atau sama dengan 65 tahun seharga 18 dollar



- Dan selain ketiga kategori diatas harganya normal yaitu 23 dollar
- Program harus menggunakan perulangan untuk menghitung total keseluruhan harga yang diinputkan secara berulang, dan juga jika uang yang diinputkan berlebih maka program wajib mengembalikan nilai kembalian uang tersebut.

Input Process Output

Input:

1. Umur pengunjung (dari pengguna, dalam tahun)
 - Dapat dimasukkan berulang kali sampai pengguna menginput -1 untuk berhenti.
2. Uang yang dibayarkan (dari pengguna, dalam dollar)

Proses:

1. Menentukan harga tiket berdasarkan umur menggunakan fungsi `hitung_tiket(umur)`:
 - Jika $\text{umur} \leq 2$: Harga = \$0 (gratis)
 - Jika $3 \leq \text{umur} \leq 12$: Harga = \$14
 - Jika $\text{umur} \geq 65$: Harga = \$18
 - Jika tidak termasuk kategori di atas: Harga = \$23
2. Menjumlahkan total harga tiket untuk semua pengunjung yang diinput.
3. Meminta pengguna untuk memasukkan uang yang dibayarkan.
4. Menghitung kembalian jika uang yang dibayarkan lebih besar dari total harga tiket:
 - $\text{Kembalian} = \text{Uang yang dibayarkan} - \text{Total harga tiket}$

Output:

1. Harga tiket untuk setiap pengunjung berdasarkan umur yang dimasukkan.
2. Total keseluruhan harga tiket untuk semua pengunjung.
3. Kembalian uang jika ada, setelah pembayaran dilakukan.



Pseudocode

def variabel hitung_umur

- Jika umur kurang dari atau sama dengan 2 tahun, maka fungsi akan mengembalikan nilai 0 (gratis).
- Jika umur antara 3 hingga 12 tahun, fungsi mengembalikan nilai 14 (harga tiket untuk anak-anak).
- Jika umur 65 tahun ke atas, fungsi mengembalikan nilai 18 (harga tiket untuk lansia).
- Jika umur tidak termasuk dalam kategori di atas, fungsi mengembalikan nilai 23 (harga tiket normal).

def main() : menginisialisasi variabel total_harga menjadi 0 untuk menyimpan total harga tiket yang harus dibayar.

Menggunakan perulangan WHILE, program meminta pengguna untuk memasukkan umur pengunjung. Jika pengguna memasukkan -1, program akan berhenti meminta input.

Untuk setiap umur yang dimasukkan, program akan memanggil fungsi hitung_tiket(umur) untuk menentukan harga tiket berdasarkan umur tersebut.

Harga tiket kemudian ditambahkan ke total_harga, dan program akan menampilkan harga tiket untuk umur yang dimasukkan.

Setelah semua umur dimasukkan, program menampilkan total keseluruhan harga tiket.

Program kemudian menggunakan perulangan WHILE lagi untuk meminta pengguna memasukkan uang yang dibayarkan.

Jika uang yang dimasukkan kurang dari total harga tiket, program akan memberi tahu pengguna dan meminta input uang lagi.

Jika uang cukup, program akan menghitung kembalian dengan mengurangi total harga tiket dari uang yang dibayarkan dan menampilkan jumlah kembalian tersebut. memanggil fungsi main() untuk menjalankan seluruh program.



Source Code

```
def hitung_tiket(umur):
    if umur <= 2:
        return 0
    elif 3 <= umur <= 12:
        return 14
    elif umur >= 65:
        return 18
    else:
        return 23

def main():
    total_harga = 0

    while True:
        try:
            umur = int(input("Masukkan umur pengunjung (atau -1 untuk selesai): "))
            if umur == -1:
                break

            harga_tiket = hitung_tiket(umur)
            total_harga += harga_tiket
            print(f"Harga tiket untuk umur {umur} tahun adalah: ${harga_tiket}")
```

```
        except ValueError:
            print("Input tidak valid. Harap masukkan angka yang benar.")

    print(f"Total keseluruhan harga tiket: ${total_harga}")

    while True:
        try:
            uang = float(input("Masukkan uang yang dibayarkan: "))
            if uang < total_harga:
                print("Uang yang dibayarkan kurang dari total harga tiket. Silakan masukkan lagi.")
            else:
                kembalian = uang - total_harga
                print(f"Kembalian uang Anda adalah: ${kembalian:.2f}")
                break
        except ValueError:
            print("Input tidak valid. Harap masukkan angka yang benar.")

if __name__ == "__main__":
    main()
```



Output

```
Masukkan umur pengunjung (atau -1 untuk selesai): 23
Harga tiket untuk umur 23 tahun adalah: $23
Masukkan umur pengunjung (atau -1 untuk selesai): 65
Harga tiket untuk umur 65 tahun adalah: $18
Masukkan umur pengunjung (atau -1 untuk selesai): 40
Harga tiket untuk umur 40 tahun adalah: $23
Masukkan umur pengunjung (atau -1 untuk selesai): 2
Harga tiket untuk umur 2 tahun adalah: $0
```

4. File Praktikum

Github Repository:

5. Soal Latihan

Soal:

1. Apa yang dimaksud dengan kondisi percobaan bersarang dan seperti apa contoh implementasinya?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. Ini berguna saat kita membutuhkan lebih dari satu kondisi untuk menentukan alur eksekusi program
2. Program meminta pengguna memasukkan umur, yang disimpan di variabel age. Program mengecek kondisi if:
 - Jika agekurang dari 12, maka kategori "Anak-anak" dan harga Rp10.000 dipilih.
 - Jika ageantara 12 hingga 18, maka kategori "Remaja" dan harga Rp15.000 dipilih.
 - Jika agelebih dari 18, maka kategori "Dewasa" dan harga Rp25.000 dipilih.Setelah itu, program mencetak kategori dan harga tiket yang harus dibayar



6. Kesimpulan

- Dalam pengerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
- Kita dapat mengetahui Struktur Kendali itu adalah menganalisis data yang menggunakan bahasa pemrograman python, yaitu perulangan dengan while dan for. Perulangan for disebut counted loop (perulangan yang terhitung), sementara perulangan while disebut uncounted loop (perulangan yang tak terhitung)

7. Cek List (✓)

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

8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	60 Menit	Menarik
2.	Latihan Kedua	60 Menit	Menarik

Keterangan:

- Menarik
- Baik
- Cukup
- Kurang

