

**LAPORAN RESMI  
MODUL III  
LOOPING (PERULANGAN)  
ALGORITMA PEMROGRAMAN**



NAMA	:	MUHAMAD DIMAS FEBRIAN
N.R.P	:	230441100035
DOSEN	:	MOHAMMAD SYARIEF, S.T.,M.Cs.
ASISTEN	:	KUKUH COKRO WIBOWO
TGL PRAKTIKUM	:	6 OKTOBER 2023

Disetujui : 11 Oktober 2023  
Asisten

**KUKUH COKRO WIBOWO**  
**21.04.411.00102**



**LABORATORIUM BISNIS INTELIJEN SISTEM  
PRODI SISTEM INFORMASI  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS TRUNOJOYO MADURA**

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Pemograman merupakan proses menulis, menguji, dan memelihara kode instruksi yang memberikan perintah atau instruksi komputer untuk melakukan suatu operasi atau tugas tertentu. Kode ini ditulis menggunakan bahasa pemograman yang berupa kumpulan instruksi yang dapat dipahami oleh komputer. Bahasa pemograman juga banyak sekali jenisnya, salah satunya adalah Python. Python sendiri dikembangkan pertama kali oleh seorang programmer asal Belanda bernama Guido Van Rossum pada akhir 1980-an yang awal nya diciptakan untuk mengatasi kekurangan dari bahasa pemograman ABC yang ia gunakan sebelumnya. Nama Python sendiri diambil dari acara komedi televisi inggris yang disukai oleh Guido, yaitu “Monty Python’s Flying Circus”. Nama ini ia cetuskan untuk memberikan nama yang unik dan mudah diingat. Salah satu aspek yang membedakan Python adalah fokusnya pada keterbacaan kode atau *readability*, bahasa python sendiri dirancang untuk membuat kode mudah dibaca dan dipahami.

Pada modul ketiga yang membahas tentang Looping (pengulangan) yang ada dalam bahasa pemograman. Looping (Perulangan) sendiri adalah program yang memungkinkan untuk mengulangi serangkaian pernyataan atau tindakan berkali-kali. Penulangan juga memungkinkan untuk mengeksekusi suatu pernyataan tanpa harus menulis ulang program berkali-kali.

#### **1.2 Tujuan**

- Mampu memahami penggunaan statement perulangan dan mampu membuat program dengan menerapkan konsep perulangan.

## **BAB II**

### **DASAR TEORI**

#### **2.1 Pengertian Perulangan**

Perintah perulangan di gunakan untuk mengulang pengeksekusian statemen-statemen hingga berkali-kali sesuai dengan iterasi yang diinginkan. Dalam python, perintah untuk perulangan (looping) adalah while dan for.

#### **2.2 Perintah While**

Perintah While pada python merupakan perintah yang paling umum digunakan untuk proses iterasi. Konsep sederhana dari perintah while adalah ia akan mengulang mengeksekusi statemen dalam blok while selama nilai kondisi benar. Dan ia akan keluar atau tidak melakukan eksekusi blok statemen jika nilai kondisinya salah.

#### **2.3. Perintah For**

Perintah for dalam python mempunyai ciri khas tersendiri dibandingkan dengan bahasa pemrograman lain. Tidak hanya mengulang bilang-bilangan sebuah ekspresi aritmatik, atau memberikan keleluasan dalam mendefinisikan iterasi perulangan dan menghentikan perulangan pada saat kondisi tertentu. Dalam python statemen for bekerja mengulang berbagai macam tipe data sekuisial seperti list, string, dan tuple.

## **2.4. Perintah Break, Continue, Dan Pass**

### **2.4.1 Perintah Break**

Perintah Break, Digunakan untuk menghentikan jalannya proses iterasi pada statemen for atau while. Statemen yang berada di bawah break tidak akan di eksekusi dan program akan keluar dari proses looping.

### **2.4.2 Perintah Continue**

Perintah Continue, menyebabkan alur program kembali ke perintah looping. Jadi jika dalam sebuah perulangan terdapat statemen continue, maka program akan kembali ke perintah looping untuk iterasi selanjutnya.

### **2.4.3 Perintah Pass**

Perintah Pass, mengakibatkan program tidak melakukan apa-apa. Perintah pass biasanya digunakan untuk mengabaikan suatu blok statemen perulangan, pengkondisian, class, dan fungsi yang belum didefinisikan badan programnya agar tidak terjadi error ketika proses compilasi.

## BAB III

### TUGAS PENDAHULUAN

#### 3.1 Soal

1. Jelaskan pengertian dan fungsi dari Continue, Break, dan Perintah pass!
2. Buatlah contoh soal sederhana dengan menggunakan perulangan for dan while!
3. Ada berapa macam bentuk perulangan, apa perbedaanya?

#### 3.2 Jawaban

1. Perintah Break, digunakan untuk menghentikan jalannya proses iterasi pada statemen for atau while. Statemen yang berada di bawah break tidak akan di eksekusi dan program akan keluar dari proses looping.  
Perintah Continue, menyebabkan alir program kembali ke perintah looping. Jadi jika dalam sebuah perulangan terdapat statemen continue, maka program akan kembali ke perintah looping untuk iterasi selanjutnya.  
Perintah Pass, menyatakan program tidak melakukan tindakan apa-apa. Perintah pass biasanya digunakan untuk mengabaikan suatu blok statemen perulangan, pengkondisian, class, dan fungsi yang belum didefinisikan badan programnya agar tidak terjadi error ketika proses compilasi.
2. for : Mencetak angka dari 1 sampai 10

```
for i in range (1,11):  
    print (i)
```

while : Mencetak angka ganjil diantara 1 sampai 10

```
i = 1  
while i < 10:  
    print (i)  
    i = i+2
```

3. Dalam bahasa pemrograman python terdapat dua macam bentuk perulangan yaitu for dan while . while memiliki konsep sederhana yaitu akan mengulang eksekusi statemen selama nilai kondisinya benar dan tidak akan melakukan eksekusi jika nilai kondisinya salah. Sedangkan for , dapat mengulang bilangan-bilangan secara ekspresi aritmatik , dan memberikan keleluasaan dalam mendefinisikan iterasi perulangan dan menyentuhkan perulangan pada saat kondisi tertentu.

## **BAB IV**

### **IMPLEMENTASI**

#### **4.1 Perulangan Dengan Bentuk NIM**

##### **4.1.1. Source Code**

```
ukuran = int(input("Masukkan Ukuran : "))
print(" ")

if ukuran >= 1:
    print()
else:
    print("invalid")

#angka pertama
for i in range(ukuran):
    if i == 0 or i == ukuran-1:
        print("x" * ukuran)
    else:
        print("x" + " " * (ukuran-2) + "x")
print(" ")

#angka kedua
for j in range(ukuran):
    if j == 0 or j == ukuran-1 or j == ukuran//2 :
        print("x" * ukuran)
    else:
        print(" " * (ukuran-1)+ "x")
print(" ")

#angka ketiga
for j in range(ukuran):
    if j == 0 or j == ukuran-1 or j == ukuran//2 :
        print("x" * ukuran)
    elif j > 0 and j < ukuran//2:
        print("x")
    else:
        print(" " * (ukuran-1) + "x")
```

#### **4.1.2 Penjelasan**

Pada penugasan pertama, yaitu membuat perulangan berbentuk NIM. Pertama, user di minta untuk menginput ukuran yang di inginkan lalu langsung akan muncul tiga angka terakhir NIM. Untuk angka yang pertama yaitu menggunakan perulangan for dengan jarak yaitu input dari user di awal lalu menggunakan pengkondision untuk membuat garis atas dan samping nya untuk membentuk angka 0. Untuk angka kedua, juga memakai cara yang hampir sama tetapi untuk menciptakan garis di tengah maka menggunakan rumus input pengguna dibagi dua dan memberikan print spasi di bagian kanan hingga tengah untuk menjadi angka 3. Untuk angka ketiga, menggunakan skrip yang mirip angka tiga untuk menciptakan garis atas, tengah, dan bawah. Lalu menggunakan print huruf x lalu spasi sebanyak inputan di samping atas kanan dan print spasi sebanyak inputan lalu x di samping bawah kiri sehingga berbentuk huruf 5.

#### **4.1.3 Hasil**

```
Masukkan Ukuran : 7

xxxxxx
x   x
x   x
x   x
x   x
x   x
xxxxxx

xxxxxx
x
x
xxxxxx
x
x
xxxxxx

xxxxxx
x
x
xxxxxx
x
x
xxxxxx
```

## 4.2 Perulangan Bilangan Prima

### 4.2.1 Source Code

```
angka_awal = int(input("Masukkan Angka Awal : "))
angka_akhir = int(input("Masukkan Angka Akhir : "))

for i in range(angka_awal, angka_akhir+1):
    if i > 1:
        for j in range(2, i):
            if (i % j) == 0:
                break
        else:
            print(i)
```

### 4.2.2. Penjelasan

Pada penugasan kedua, yaitu membuat program perulangan yang menampilkan bilangan prima. Pada saat memulai program user diminta untuk menginputkan dua input bertipe integer yaitu angka awal dan angka akhir. Lalu angka awal tersebut dimasukan kedalam variable i yang dapat menampung seluruh angka yang ada di input angka awal sampai dengan input angka akhir, di bagian input angka akhir ditambahkan “+1” supaya program membaca hingga angka akhir dan bukan dibaca sebagai sebelum angka akhir.

Lalu masuk dalam pengkondisian yang dimana terdapat syarat mutlak dari bilangan prima yaitu angka yang lebih dari satu. Lalu dilanjutkan kedalam variable j yang dimana menampung angka 2 dan titik akhir nya adalah titik akhir dalam variable I yaitu input angka akhir dari user. Lalu masuk dalam pengkondisian dan semua angka dalam variable i akan di bagi dengan semua angka dalam variable j dan jika hasil yang keluar adala 0 maka akan di hentikan dengan *break* dan angka dalam variabel I yang bukan modulus nya 0 akan di print sebagai hasil.

#### 4.2.3. Hasil

```
Masukkan Angka Awal : 1
Masukkan Angka Akhir : 40
2
3
5
7
11
13
17
19
23
29
31
37
```

### 4.3 Program Penghitungan Denda

#### 4.3.1. Source Code

```
while True:
    waktupinjam = int(input("Berapa Hari anda meminjam buku :"))
    DendaHarian = 2000
    DendaMingguan = 5000
    Total = 0

    for i in range (1,waktupinjam+1) :
        if i > 7 :
            Total += DendaHarian
        if i % 7 == 0 :
            Total += DendaMingguan
    if waktupinjam > 7 :
        print(f"Anda Meminjam buku lebih dari 1 Minggu Anda
Terlambat Mengembalikan Buku selama {waktupinjam-7} Hari Dan
Anda terkena Denda Sebesar Rp {Total} ")

    else :
        print(f"Anda Meminjam Buku Selama {waktupinjam} hari,
Jadi Anda Tidak Terkena Denda!")

    ulangi = input("Apakah ingin mengulangi program?
(ya/tidak): ")
    if ulangi != "ya":
        break
```

#### **4.3.2. Penjelasan**

Pada penugasan ketiga, yaitu membuat program penghitungan denda jika meminjam buku lebih dari seminggu. Menggunakan perulang while true yang dimana awal nya pengguna menginputkan berapa hari pengguna meminjam buku jika lebih dari seminggu akan dikenakan denda 2000 perharinya dan perminggu nya akan di kenakan denda tambahan sebanyak 5000 jika masih belum mengembalikan juga. Menggunakan variabel i yang menampung angka satu dan titik akhirnya yaitu variable input pengguna ditambah dengan satu agar terbaca program. Lalu pengkondisian jika i lebih dari 7 hari maka tiap angka di atas 7 akan di kenakan denda 2000 dan ditambahkan ke variable total denda. Dalam pengkondisian kedua jika didalam variabel i terdapat kelipatan 7 selain angka 7 itu sendiri maka akan di kenakan denda mingguan sebanyak 5000 dan akan di tambahkan dalam variabel total denda.lalu output nya jika waktu pinjam lebih besar dari 7 hari maka akan di print pernyataan “anda telat” beserta total denda yang dikenakan. Dan jika tidak maka akan keluar output anda tidak terkena denda. Dan terakhir membuat fitur mengulangi program atau tidak dengan cara membuat variabel input string yang di dalamnya terdapat kata ya atau tidak. Dan membuat pengkondisian jika variabel input pengguna sama dengan variable yaitu “ya” maka perulangan akan berhenti dan akan di mulai dari awal. Dan jika tidak maka semua program akan berhenti.

#### **4.3.3. Hasil**

```
Berapa Hari anda meminjam buku : 44  
Anda Meminjam buku lebih dari 1 Minggu Anda Terlambat Mengembalikan Buku selama 37 Hari Dan Anda terkena Denda Sebesar Rp 99000  
Apakah ingin mengulangi program? (ya/tidak): tidak  
selesai
```

## **BAB V**

### **PENUTUP**

#### **5.1 Analisa**

Dari hasil praktikum kali ini praktikan dapat mempelajari cara menerapkan perulangan while dan for. Praktikan juga dapat memahami penggunaan tanda operasi seperti pembagian bulat tanda “`+=`” atau tanda “`!=`” yang dimana berguna dalam membuat program.

Perulangan (looping) adalah salah satu elemen dasar yang sangat penting dalam ilmu pemrograman. Perulangan dapat memungkinkan pengulangan eksekusi serangkaian pernyataan atau statement berulang kali berdasarkan kondisi tertentu.

Perulangan dapat memudahkan dalam mengeksekusi suatu pernyataan atau statement yang harus ditulis secara berulang-ulang menjadi hanya perlu menulis suatu program perulangan dan dapat ditentukan berapa kali akan diulang.

Perulangan juga menjadi ilmu penting dalam mempelajari pemograman, tidak hanya bahasa python saja tapi semua bahasa pemograman bisa menerapkan perulangan dengan *command* nya tersendiri. Dari hasil praktikum ini juga praktikan dapat menganalisa bahwa perulangan memiliki banyak kegunaan, terutama dalam sebuah studi kasus yang melibatkan eksekusi suatu statement berulang-ulang.

#### **5.2 Kesimpulan**

Dari materi dan latihan soal yang terdapat pada laporan praktikum ini. Kita dapat mempelajari dan mempraktikan penggunaan sebuah program perulangan pada bahasa pemograman python. Seperti contoh diatas, kita dapat menerapkan program perualangan dalam sebuah program komputer dengan menggunakan bahasa pemograman python.

