

LAPORAN RESMI
MODUL III
PERULANGAN (LOOPING)
ALGORITMA PEMROGRAMAN



NAMA	: KEVIN MALIK FAJAR
N.R.P	: 200441100014
DOSEN	: IMAMAH, S.Kom., M.Kom.
ASISTEN	: NOVI LIANA
TGL PRAKTIKUM	: 17 November 2020

Disetujui : 24 November 2020

Asisten

NOVI LIANA
190441100087



LABORATORIUM BISNIS INTELIJEN SISTEM
JURUSAN SISTEM INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS TRUNOJOYO MADURA

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perintah perulangan di gunakan untuk mengulang pengeksekusian statemenstatemen hingga berkali-kali sesuai dengan iterasi yang diinginkan. Dalam python perintah untuk perulangan (loop) adalah while dan for. 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 kondisinya benar. Dan ia akan keluar atau tidak melakukan eksekusi blok statemen jika nilai kondisinya salah. Perintah for dalam python mempunyai ciri khas tersendiri dibandingkan dengan bahasa pemrograman lain. Tidak hanya mengulang bilangan-bilangan sebuah ekspresi aritmatik, atau memberikan keleluasaan dalam mendefinisikan iterasi perulangan dan menghentikan perulangan pada saat kondisi tertentu. Dalam python, statemen for bekerja mengulang berbagai macam tipe data sekuensial seperti List, String, dan Tuple.

Python diciptakan oleh Guido van Rossum pertama kali di *Centrum Wiskunde & Informatica* (CWI) di Belanda pada awal tahun 1990-an. Bahasa python terinspirasi dari bahasa pemrograman ABC. Sampai sekarang, Guido masih menjadi penulis utama untuk python, meskipun bersifat open source sehingga ribuan orang juga berkontribusi dalam mengembangkannya. Di tahun 1995, Guido melanjutkan pembuatan python di *Corporation for National Research Initiative* (CNRI) di Virginia Amerika, di mana dia merilis beberapa versi dari python. Pada Mei 2000, Guido dan tim Python pindah ke BeOpen.com dan membentuk tim BeOpen PythonLabs. Di bulan Oktober pada tahun yang sama, tim python pindah ke Digital Creation (sekarang menjadi Perusahaan Zope). Pada tahun 2001, dibentuklah Organisasi Python yaitu Python Software Foundation (PSF). PSF merupakan organisasi nirlaba yang dibuat khusus untuk semua hal yang berkaitan dengan hak intelektual Python. Perusahaan Zope menjadi anggota sponsor dari PSF.

1.2 Tujuan

Mampu memahami dan menjelaskan macam – macam perulangan (*looping*) yang terdapat pada Algoritma Pemrograman.

BAB II

DASAR TEORI

2. 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 (loop) 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 kondisinya benar. Dan ia akan keluar atau tidak melakukan eksekusi blok statemen jika nilai kondisinya salah.

Bentuk umum statemen <i>while</i> , while (kondisi) : statemen
--

Pada contoh 1, merupakan contoh sederhana penggunaan while. Pada contoh di atas program akan terus mengeksekusi statemen dalam badan while, dikarenakan kondisinya selalu benar (true). Kondisi seperti ini disebut infinite loop.

Contoh 1 :

1	x = "Wuland"
2	while x:
3	
4	print (x)
5	x = x[1:]

Contoh 2 :

1	a = 0; b = 10
2	while a < b :
3	print (a)
4	a = a + 1

2.3 Perintah For

Perintah for dalam python mempunyai ciri khas tersendiri dibandingkan dengan bahasa pemrograman lain. Tidak hanya mengulang bilangan-bilangan sebuah ekspresi aritmatik, atau memberikan keleluasaan dalam mendefinisikan iterasi perulangan dan menghentikan perulangan pada saat kondisi tertentu. Dalam python, statemen for bekerja mengulang berbagai macam tipe data sekuensial seperti List, String, dan Tuple.

```
Bentuk umum perintah for,
for (variabel) in (objek) :
    statemen
else:
    statemen
```

Contoh sederhana menggunakan perintah for:

```
1 for i in [5, 4, 3, 2, 1]:
2     print (i)
```

Pada contoh 2, perintah perulangan terjadi dimana data-data untuk iterasi (objek) berada dalam List. Jadi elemen-elemen yang berada dalam List akan di masukkan (assign) ke dalam variabel target yaitu i.

Contoh 2 :

```
1 T = [(1,2), (3,4), (5,6)]
2 for (a,b) in T:
3     print (a,b)
```

Pada contoh 3, merupakan penggunaan tipe data Tuple untuk proses perulangan. Elemen pada tuple akan di assign kedalam variabel a dan b.

```
1 nama = ['budi', 'andi', 'rudi', 'sandi']
2 usia = [20, 18, 22, 19]
3 for i in range(len(nama)):
4     print (nama[i], ' berusia ', usia[i], ' tahun' )
```

2.4 Perintah Break, Continue dan Pass

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.

Contoh break :

```
1 x = 4
2 while x < 5:
3     if x == 3:
4         break
5     print(x)
6     x = x+1
7 else:
8     print("Loop sdh selesai dikerjakan")
```

Perintah Continue

Statemen 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.

Contoh continue :

```
1 n = 10
2 while n:
3     n = n - 1
4     if n % 2 != 0:
5         continue
6     print(n)
```

Perintah Pass

Statemen pass mengakibatkan 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 kompilasi.

Contoh program :

```
1 #program tidak akan melakukan
2 # proses looping
3 # while True : pass
4 while True : pass
```

BAB III

TUGAS PENDAHULUAN

BAB III

TUGAS PENDAHULUAN

3.1 Pertanyaan

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 bedanya?

3.2 Jawaban

1. - Statement continue menyebabkan alur Program kembali ke Perintah looping. Jadi jika dalam sebuah Perulangan terdapat Statement Continue, maka Program akan kembali ke Perintah looping untuk iterasi selanjutnya.

- Statement 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.

- Statement Pass mengakibatkan 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 kompilasi.

2. - Buatlah Program Pendeteksi bilangan Ganjil dan Genap di Range tertentu, jika Input 1 maka akan mendeteksi Ganjil dan jika 2 maka akan mendeteksi Genap, selain itu error.

- Buatlah Program Pendeteksi bilangan Ganjil dan Genap di Range tertentu, jika Input 1 maka mendeteksi bilangan Ganjil dan 2 akan mendeteksi Genap, jika Input 0 maka Program diulang

dan jika input 1 maka Program akan di Stop.

```
(1) Pilihan = int(input('masukan Pilihan: 1. Ganjil | 2. Genap: '))
```

```
if Pilihan == 1:
```

```
    angka = int(input('masukan angka: '))
```

```
    for x in range(angka):
```

```
        if x % 2 == 1:
```

```
            Print(x)
```

```
elif Pilihan == 2:
```

```
    angka = int(input('masukan angka: '))
```

```
    for z in range(angka):
```

```
        if z % 2 == 0:
```

```
            Print(z)
```

```
else:
```

```
    Print('Perintah error')
```

```
(2.) z = 0
```

```
while z == 0:
```

```
    Pilihan = int(input('masukan Pilihan: 1. Ganjil | 2. Genap: '))
```

```
    angka = int(input('masukan angka: '))
```

```
    if Pilihan == 1:
```

```
        for x in range(angka):
```

```
            if x % 2 == 1:
```

```
                Print(x)
```

```
    else:
```

```
        for x in range(angka):
```

```
            if x % 2 == 0:
```

```
                Print(x)
```

```
    Perintah = int(input('Ingin Ulang Program? 0 untuk Ya | 1 untuk tidak'))
```

```
    if Perintah == 0:
```

```
        Continue
```

```
    else:
```

```
        break
```

3. Ada 2 yaitu while dan for. Perbedaannya adalah jika for digunakan dalam perulangan yang sudah diketahui jumlah perulangannya sedangkan while digunakan ketika jumlah perulangannya belum ditentukan.

BAB IV

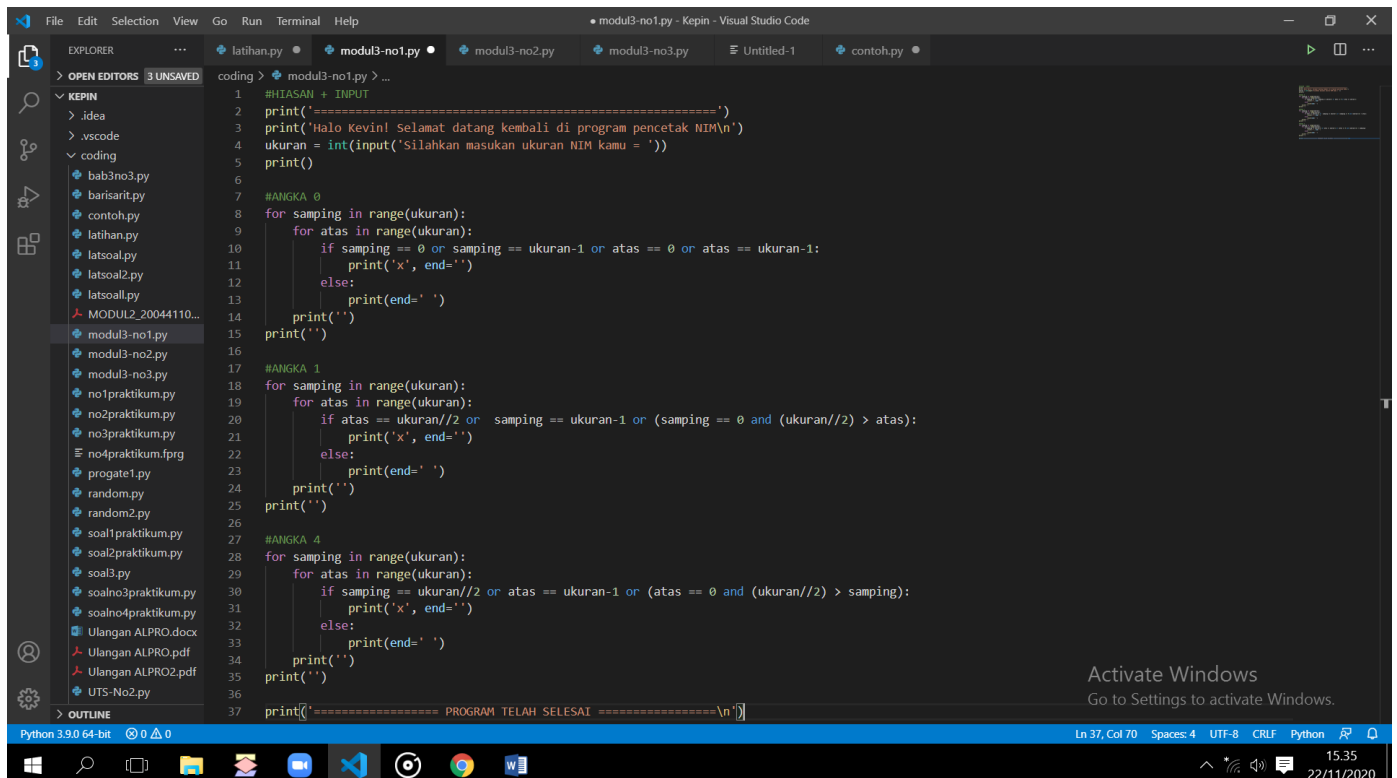
IMPLEMENTASI

4.1. Implementasi

1. Buatlah program dengan bentuk angka NIM terakhir kalian , misalnya 036!

Outputnya:

```
C:\Python34\python
Masukan Size : 6
xxxxxxx
x  x
x  x
x  x
x  x
x  x
xxxxxxx
hhhhhhh
  x
  x
hhhhhhh
  n
  n
hhhhhhh
xxxxxxx
x
x
xxxxxxx
x  x
x  x
xxxxxxx
```




```
File Edit Selection View Go Run Terminal Help
modul3-no1.py - Kepin - Visual Studio Code

EXPLORER
> OPEN EDITORS 2 UNSAVED
  > KEPIN
    > .idea
    > .vscode
    > coding
      bab3no3.py
      barisarit.py
      contoh.py
      latihan.py
      latsoal.py
      latsoal2.py
      latsoall.py
      MODUL2_200441100014_Kev...
      modul3-no1.py
      modul3-no2.py
      modul3-no3.py
      no1praktikum.py
      no2praktikum.py
      no3praktikum.py
      no4praktikum.fprg
      progate1.py
      random.py
      random2.py
      soal1praktikum.py
      soal2praktikum.py
      soal3.py
      soalno3praktikum.py
      soalno4praktikum.py
      Ulangan ALPRO.docx
      Ulangan ALPRO.pdf
      Ulangan ALPRO2.pdf
      UTS-No2.py

  > OUTLINE

TERMINAL
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\AXI00\Documents\Kepin> & C:/Users/AXI00/AppData/Local/Programs/Python/Python39/python.exe c:/Users/AXI00/Documents/Kepin/coding/modul3-no1.py

Halo Kevin! Selamat datang kembali di program pencetak NIM

Silahkan masukan ukuran NIM kamu = 10

xxxxxxxxxx
x      x
x      x
x      x
x      x
x      x
x      x
x      x
x      x
xxxxxxxxxx

xxxxxxx
x
x
x
x
x
x
x
x
xxxxxxxxxx

x  x
x  x
x  x
x  x
x  x
xxxxxxx
x
x
x
x

===== PROGRAM TELAH SELESAI =====

PS C:\Users\AXI00\Documents\Kepin>
```

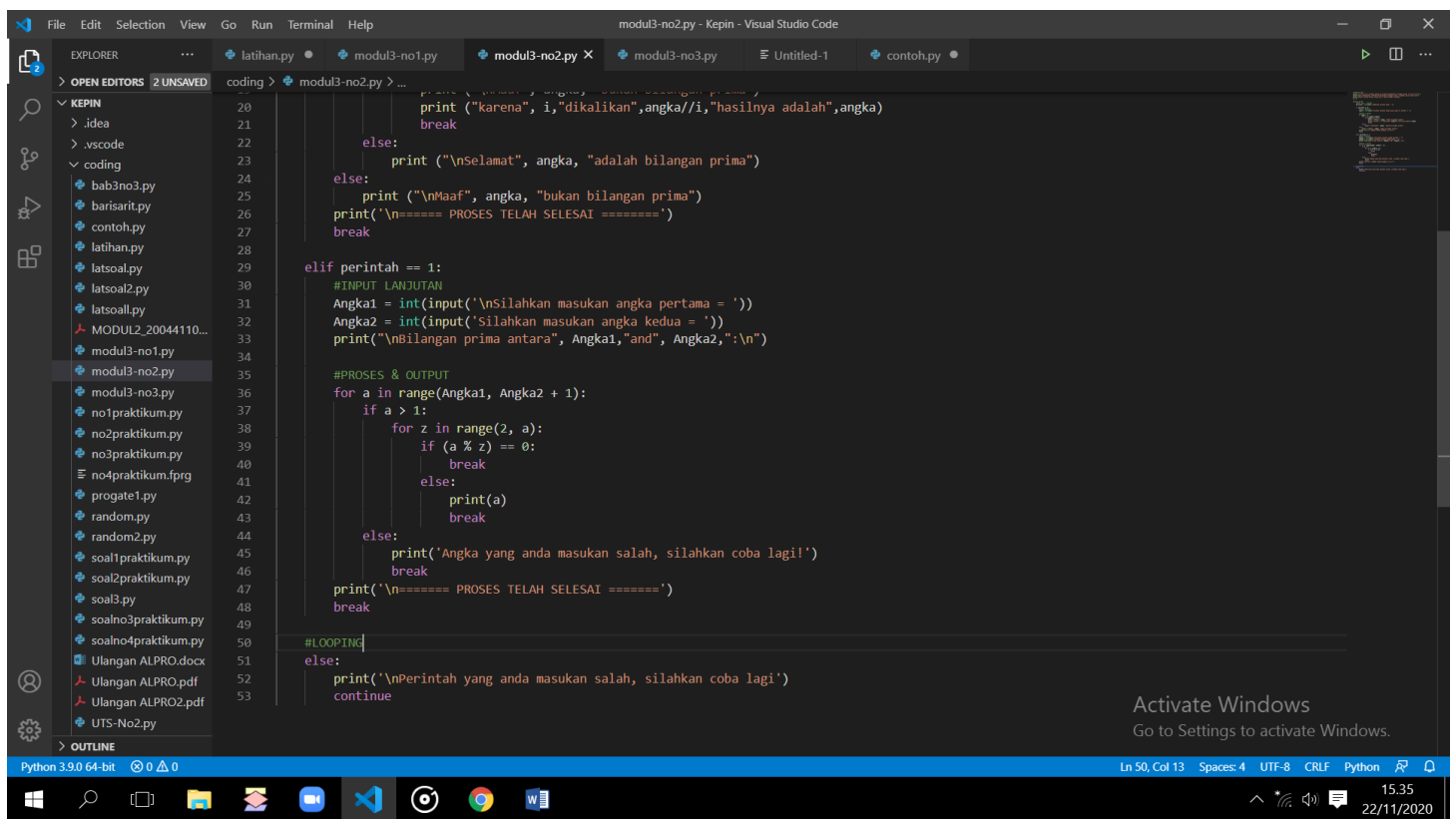
2. Buatlah program perulangan bilangan prima!

```
File Edit Selection View Go Run Terminal Help
modul3-no2.py - Kepin - Visual Studio Code

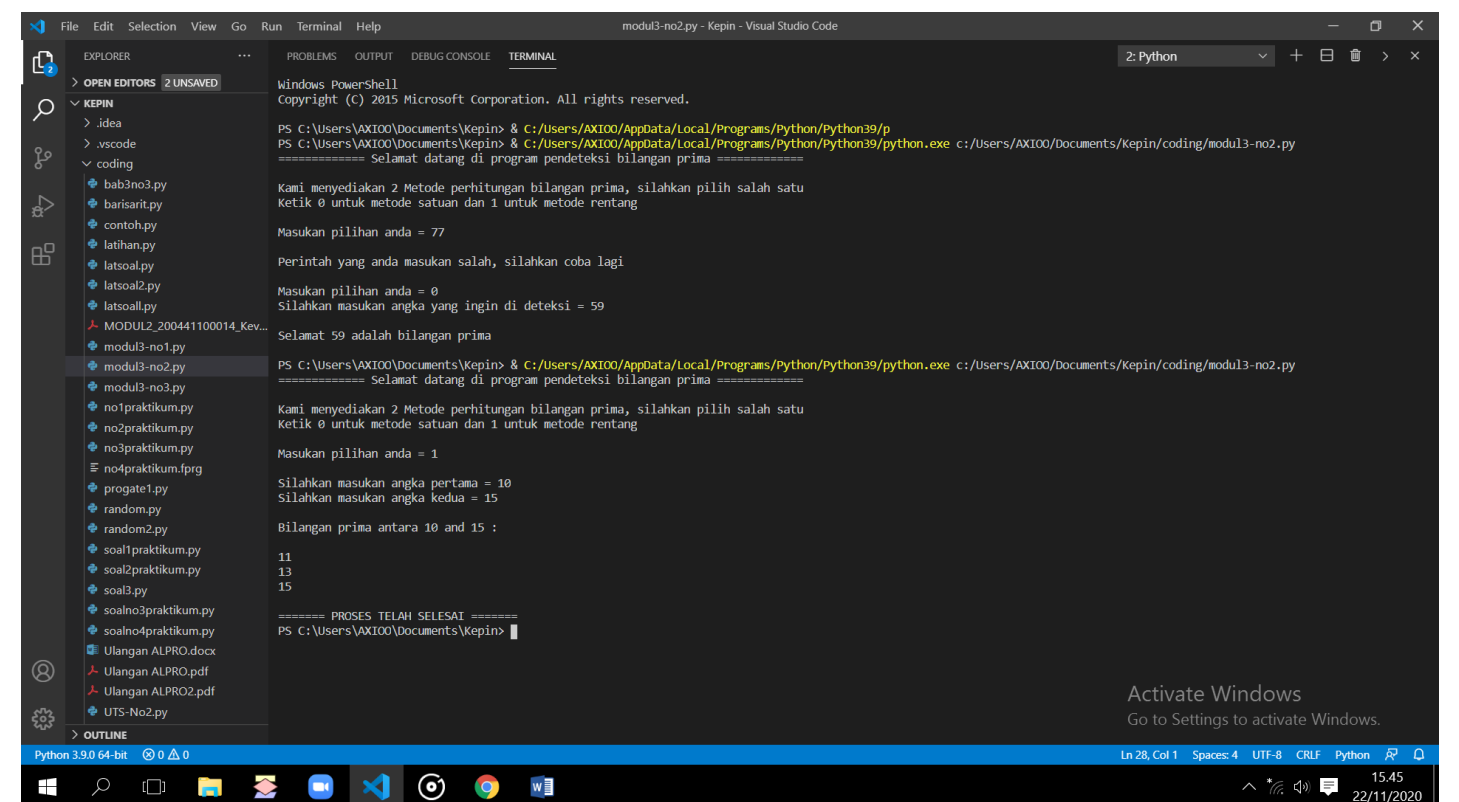
EXPLORER
> OPEN EDITORS 2 UNSAVED
  > KEPIN
    > .idea
    > .vscode
    > coding
      bab3no3.py
      barisarit.py
      contoh.py
      latihan.py
      latsoal.py
      latsoal2.py
      latsoall.py
      MODUL2_20044110...
      modul3-no1.py
      modul3-no2.py
      modul3-no3.py
      no1praktikum.py
      no2praktikum.py
      no3praktikum.py
      no4praktikum.fprg
      progate1.py
      random.py
      random2.py
      soal1praktikum.py
      soal2praktikum.py
      soal3.py
      soalno3praktikum.py
      soalno4praktikum.py
      Ulangan ALPRO.docx
      Ulangan ALPRO.pdf
      Ulangan ALPRO2.pdf
      UTS-No2.py

  > OUTLINE

modul3-no2.py
1  #TAMPILAN AWAL
2  print('===== Selamat datang di program pendeteksi bilangan prima =====\n')
3  print('Kami menyediakan 2 Metode perhitungan bilangan prima, silahkan pilih salah satu')
4  print('Ketik 0 untuk metode satuan dan 1 untuk metode rentang')
5  y = 0
6
7  while y == 0:
8      #INPUT AWAL + LOOPING
9      perintah = int(input('\nMasukan pilihan anda = '))
10
11     if perintah == 0:
12         #INPUT LANJUTAN
13         angka = int(input('Silahkan masukan angka yang ingin di deteksi = '))
14
15         #PROSES & OUTPUT
16         if angka > 1:
17             for i in range(2, angka):
18                 if (angka % i) == 0:
19                     print("\nMaaf", angka, "bukan bilangan prima")
20                     print("karena", i, "dikalikan", angka//i, "hasilnya adalah", angka)
21                     break
22             else:
23                 print("\nSelamat", angka, "adalah bilangan prima")
24         else:
25             print("\nMaaf", angka, "bukan bilangan prima")
26         print('\n===== PROSES TELAH SELESAI =====')
27         break
28
29     elif perintah == 1:
30         #INPUT LANJUTAN
31         Angka1 = int(input('\nSilahkan masukan angka pertama = '))
32         Angka2 = int(input('Silahkan masukan angka kedua = '))
33         print("\nBilangan prima antara, Angka1, and, Angka2,": "\n")
34
35         #PROSES & OUTPUT
36         for a in range(Angka1, Angka2 + 1):
37             if a > 1:
```



```
modul3-no2.py
20 print ("karena", i,"dikalikan",angka//i,"hasilnya adalah",angka)
21 break
22 else:
23     print ("\nSelamat", angka, "adalah bilangan prima")
24 else:
25     print ("\nMaaf", angka, "bukan bilangan prima")
26 print('\n===== PROSES TELAH SELESAI =====')
27 break
28
29 elif perintah == 1:
30     #INPUT LANJUTAN
31     Angka1 = int(input("\nSilahkan masukan angka pertama = "))
32     Angka2 = int(input("\nSilahkan masukan angka kedua = "))
33     print("\nBilangan prima antara", Angka1,"and", Angka2,"\n")
34
35     #PROSES & OUTPUT
36     for a in range(Angka1, Angka2 + 1):
37         if a > 1:
38             for z in range(2, a):
39                 if (a % z) == 0:
40                     break
41                 else:
42                     print(a)
43                     break
44             else:
45                 print('Angka yang anda masukan salah, silahkan coba lagi!')
46                 break
47     print('\n===== PROSES TELAH SELESAI =====')
48     break
49
50     #LOOPING
51     else:
52         print('\nPerintah yang anda masukan salah, silahkan coba lagi')
53         continue
```



```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\AXI00\Documents\Kepin> & C:/Users/AXI00/AppData/Local/Programs/Python/Python39/p
PS C:\Users\AXI00\Documents\Kepin> & C:/Users/AXI00/AppData/Local/Programs/Python/Python39/python.exe c:/Users/AXI00/Documents/kepin/coding/modul3-no2.py
===== Selamat datang di program pendeteksi bilangan prima =====

Kami menyediakan 2 Metode perhitungan bilangan prima, silahkan pilih salah satu
Ketik 0 untuk metode satuan dan 1 untuk metode rentang

Masukan pilihan anda = 77

Perintah yang anda masukan salah, silahkan coba lagi

Masukan pilihan anda = 0
Silahkan masukan angka yang ingin di deteksi = 59

Selamat 59 adalah bilangan prima

PS C:\Users\AXI00\Documents\Kepin> & C:/Users/AXI00/AppData/Local/Programs/Python/Python39/python.exe c:/Users/AXI00/Documents/kepin/coding/modul3-no2.py
===== Selamat datang di program pendeteksi bilangan prima =====

Kami menyediakan 2 Metode perhitungan bilangan prima, silahkan pilih salah satu
Ketik 0 untuk metode satuan dan 1 untuk metode rentang

Masukan pilihan anda = 1

Silahkan masukan angka pertama = 10
Silahkan masukan angka kedua = 15

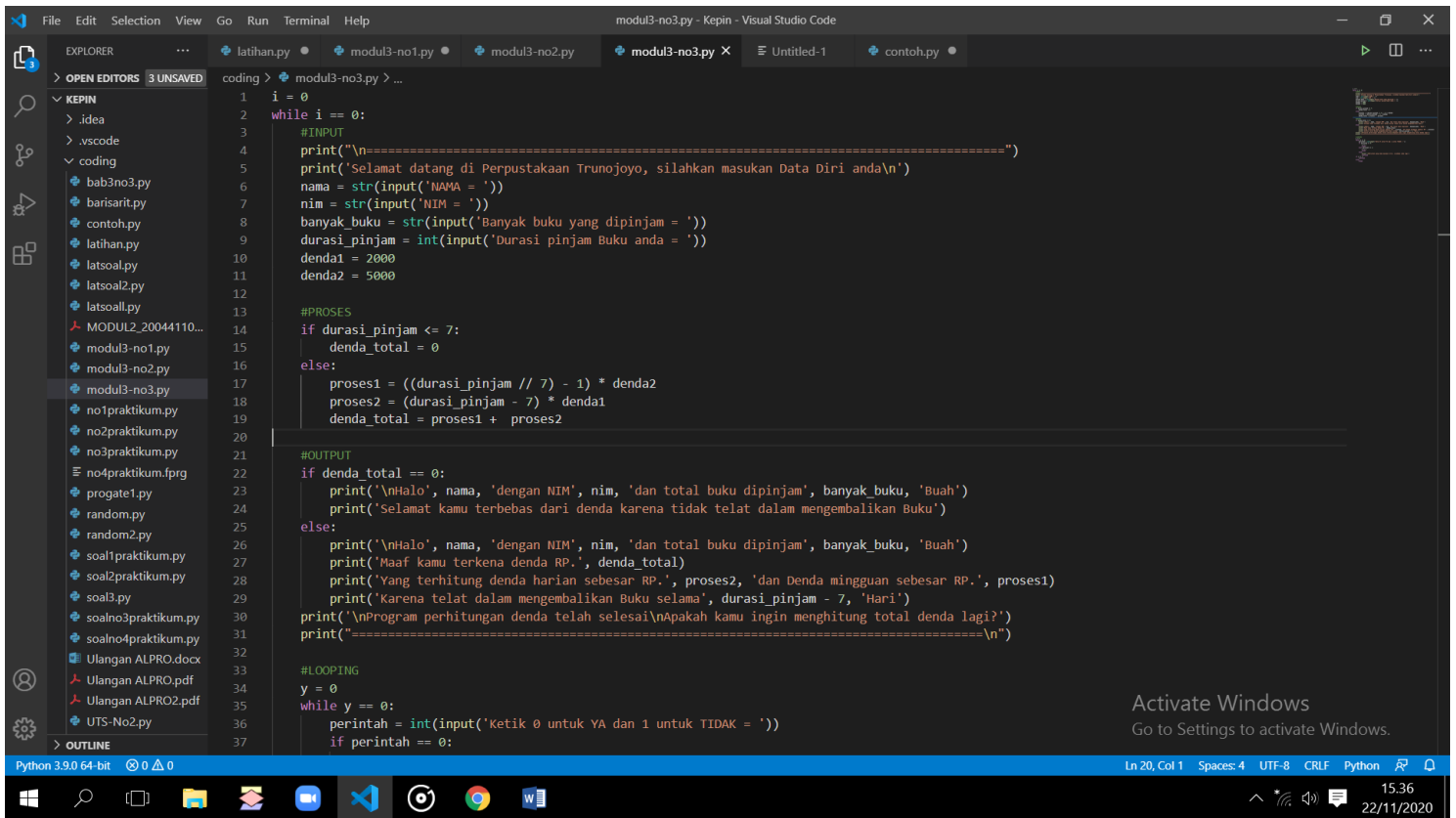
Bilangan prima antara 10 and 15 :

11
13
15

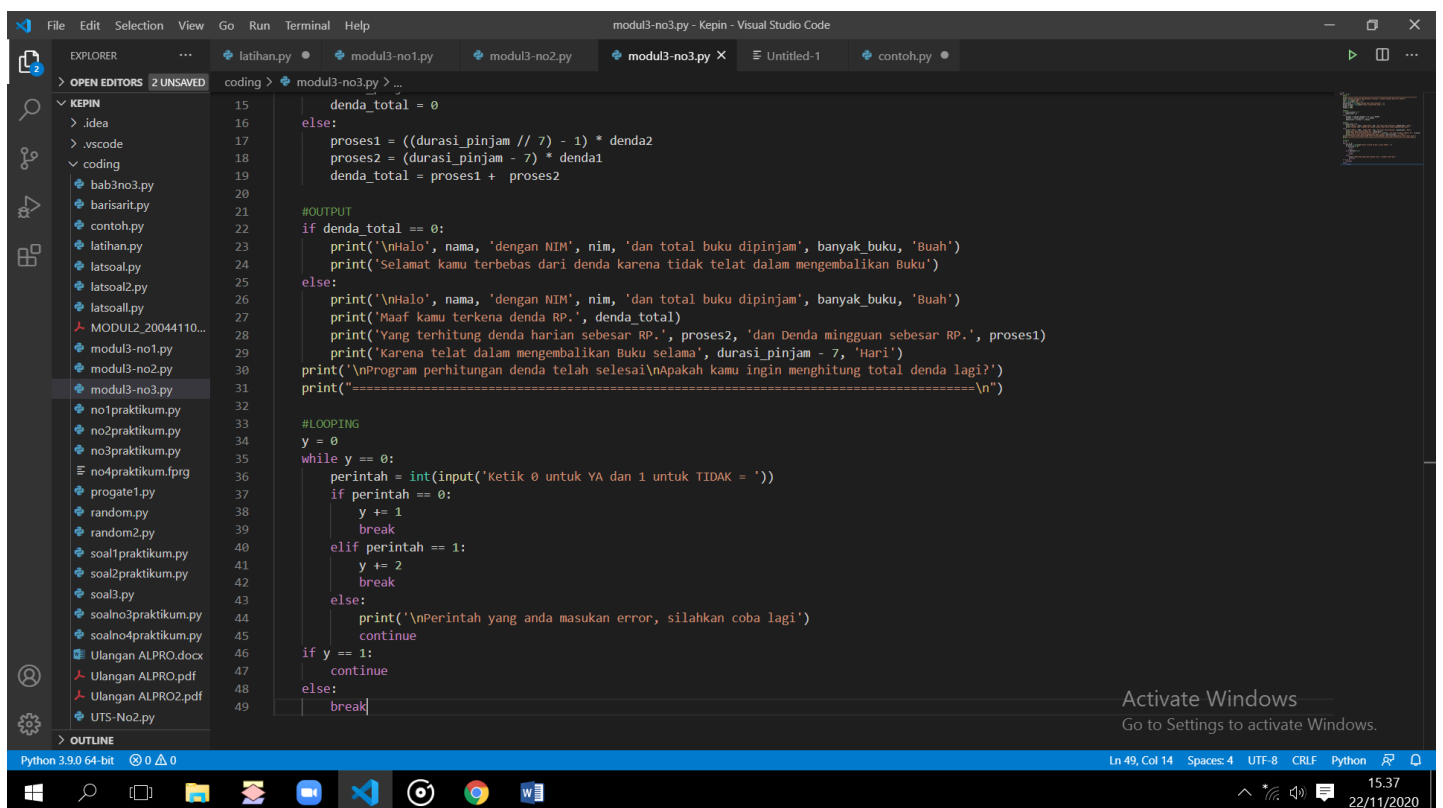
===== PROSES TELAH SELESAI =====
PS C:\Users\AXI00\Documents\Kepin>
```

- Sebuah perpustakaan melayani peminjaman dan pengembalian buku. Setiap buku hanya boleh dipinjam selama 1 minggu (7 hari). Jika lebih dari batas waktu tersebut, maka petugas perpustakaan akan memberikan denda sebanyak Rp2000 per harinya. Petugas akan memberikan denda tambahan sebesar Rp5000 setiap minggunya jika buku belum juga dikembalikan.

Bantulah petugas perpustakaan untuk membuat program penghitungan denda tersebut dengan input berupa lama peminjaman buku (dalam satuan hari) dan output denda yang harus dibayar. Setelah program selesai dijalankan, tampilkan pertanyaan menghitung lagi atau tidak. Jika dijawab “ya”, maka program akan mengulangi dari awal.



```
1 i = 0
2 while i == 0:
3     #INPUT
4     print("\n=====")
5     print('Selamat datang di Perpustakaan Trunojoyo, silahkan masukan Data Diri anda\n')
6     nama = str(input('NAMA = '))
7     nim = str(input('NIM = '))
8     banyak_buku = str(input('Banyak buku yang dipinjam = '))
9     durasi_pinjam = int(input('Durasi pinjam Buku anda = '))
10    denda1 = 2000
11    denda2 = 5000
12
13    #PROSES
14    if durasi_pinjam <= 7:
15        denda_total = 0
16    else:
17        proses1 = ((durasi_pinjam // 7) - 1) * denda2
18        proses2 = (durasi_pinjam - 7) * denda1
19        denda_total = proses1 + proses2
20
21    #OUTPUT
22    if denda_total == 0:
23        print('\nHalo', nama, 'dengan NIM', nim, 'dan total buku dipinjam', banyak_buku, 'Buah')
24        print('Selamat kamu terbebas dari denda karena tidak telat dalam mengembalikan Buku')
25    else:
26        print('\nHalo', nama, 'dengan NIM', nim, 'dan total buku dipinjam', banyak_buku, 'Buah')
27        print('Maaf kamu terkena denda RP.', denda_total)
28        print('Yang terhitung denda harian sebesar RP.', proses2, 'dan Denda mingguan sebesar RP.', proses1)
29        print('Karena telat dalam mengembalikan Buku selama', durasi_pinjam - 7, 'Hari')
30    print('\nProgram perhitungan denda telah selesai\nApakah kamu ingin menghitung total denda lagi?')
31    print("=====\n")
32
33    #LOOPING
34    y = 0
35    while y == 0:
36        perintah = int(input('Ketik 0 untuk YA dan 1 untuk TIDAK = '))
37        if perintah == 0:
```



```
15 denda_total = 0
16
17 else:
18     proses1 = ((durasi_pinjam // 7) - 1) * denda2
19     proses2 = (durasi_pinjam - 7) * denda1
20     denda_total = proses1 + proses2
21
22 #OUTPUT
23 if denda_total == 0:
24     print('\nHalo', nama, 'dengan NIM', nim, 'dan total buku dipinjam', banyak_buku, 'Buah')
25     print('Selamat kamu terbebas dari denda karena tidak telat dalam mengembalikan Buku')
26 else:
27     print('\nHalo', nama, 'dengan NIM', nim, 'dan total buku dipinjam', banyak_buku, 'Buah')
28     print('Maaf kamu terkena denda RP.', denda_total)
29     print('Yang terhitung denda harian sebesar RP.', proses2, 'dan Denda mingguan sebesar RP.', proses1)
30     print('Karena telat dalam mengembalikan Buku selama', durasi_pinjam - 7, 'Hari')
31 print('\nProgram perhitungan denda telah selesai\nApakah kamu ingin menghitung total denda lagi?')
32 print("=====\n")
33
34 #LOOPING
35 y = 0
36 while y == 0:
37     perintah = int(input('Ketik 0 untuk YA dan 1 untuk TIDAK = '))
38     if perintah == 0:
39         y += 1
40         break
41     elif perintah == 1:
42         y += 2
43         break
44     else:
45         print('\nPerintah yang anda masukan error, silahkan coba lagi')
46         continue
47 if y == 1:
48     continue
49 else:
50     break
```

modul3-no3.py - Kepin - Visual Studio Code

File Edit Selection View Go Run Terminal Help

2: Python

EXPLORER

OPEN EDITORS 2 UNSAVED

KEPIN

- .idea
- .vscode
- coding
 - bab3no3.py
 - barisarit.py
 - contoh.py
 - latihan.py
 - latsoal.py
 - latsoal2.py
 - latsoall.py
 - MODUL2_20044110...
 - modul3-no1.py
 - modul3-no2.py
 - modul3-no3.py
 - no1praktikum.py
 - no2praktikum.py
 - no3praktikum.py
 - no4praktikum.fprg
 - progate1.py
 - random.py
 - random2.py
 - soal1praktikum.py
 - soal2praktikum.py
 - soal3.py
 - soalno3praktikum.py
 - soalno4praktikum.py
 - Ulangan ALPRO.docx
 - Ulangan ALPRO.pdf
 - Ulangan ALPRO2.pdf
 - UTS-No2.py

OUTLINE

TERMINAL

```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. All rights reserved.

PS C:\Users\AXI00\Documents\Kepin> & C:/Users/AXI00/AppData/Local/Programs/Python/Python39/python.exe c:/Users/AXI00/Documents/Kepin/coding/modul3-no3.py

=====
Selamat datang di Perpustakaan Trunojoyo, silahkan masukan Data Diri anda
=====

Nama = Kevin Malik Fajar
NIM = 200441100014
Banyak buku yang dipinjam = 15
Durasi pinjam Buku anda = 20

Halo Kevin Malik Fajar dengan NIM 200441100014 dan total buku dipinjam 15 Buah
Maaf kamu terkena denda RP. 31000
Yang terhitung denda harian sebesar RP. 26000 dan Denda mingguan sebesar RP. 5000
Karena telat dalam mengembalikan Buku selama 13 Hari

Program perhitungan denda telah selesai
Apakah kamu ingin menghitung total denda lagi?
=====

Ketik 0 untuk YA dan 1 untuk TIDAK = 1
PS C:\Users\AXI00\Documents\Kepin> 
```

Activate Windows
Go to Settings to activate Windows.

Python 3.9.0 64-bit 0 0 0

Ln 49, Col 14 Spaces: 4 UTF-8 CRLF Python

15.39
22/11/2020

BAB V

PENUTUP

5.1 Analisa

Dari hasil praktikum, praktikan menganalisa bahwa Sebelum kita memulai suatu program kita harus memahami terlebih dahulu program yang akan kita buat. Seperti apa itu variabel, tipe data, dan operator selain itu kita juga harus mengetahui macam-macam perulangan yang dapat kita gunakan. Setelah kita memahaminya langsung saja memulai program yang akan dibuat. Jika program menggunakan while akan mengulang mengeksekusi statemen dalam blok while selama nilai kondisinya benar. Dan ia akan keluar atau tidak melakukan eksekusi blok statemen jika nilai kondisinya salah, perulangan while ini yang memiliki syarat dan tidak tentu berapa banyak perulangannya. Sedangkan perulangan for biasanya digunakan untuk mengulangi kode yang sudah diketahui banyak perulangannya.

.

5.2 Kesimpulan

Dari pelaksanaan praktikum ini, kita bisa lebih mempelajari Dalam bahasa pemrograman berarti memberikan perintah yang berulang kepada komputer dengan kondisi tertentu. Dalam pemrograman python terdapat 2 jenis perulangan, yaitu perulangan For dan perulangan while. Perulangan for disebut sebagai counted loop (perulangan yang terhitung) Sedangkan perulangan while disebut uncounted loop (perulangan yang tak terhitung). Perulangan for biasanya digunakan ketika kita sudah mengetahui banyak perulangannya, sedangkan perulangan while digunakan ketika kita tidak mengetahui jumlah perulangan yang akan terjadi. Namun, perulangan while dapat juga digunakan ketika kita sudah mengetahui banyak perulangannya, namun tidak sebagai ketika menggunakan perulangan for.