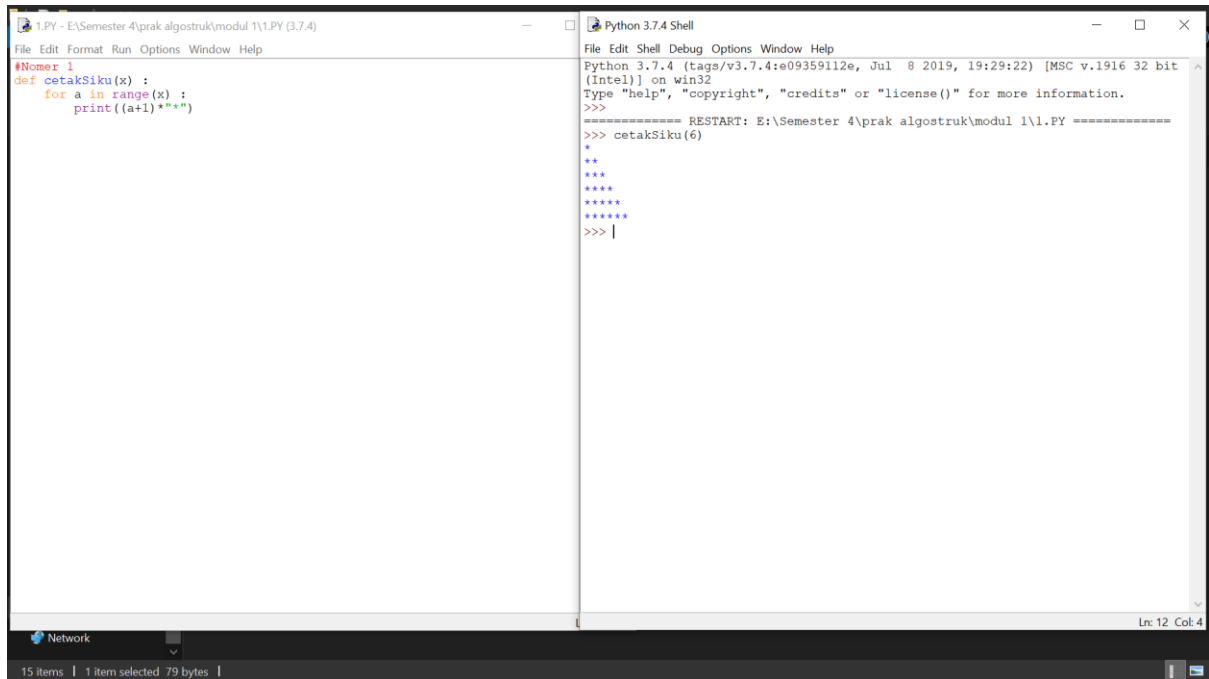


Nama : Rahmat Beny Susanto

NIM : L200180079

1.



The screenshot shows a Python 3.7.4 IDE with two windows. The left window, titled '1.PY - E:\Semester 4\prak algostruk\modul 1\1.PY (3.7.4)', contains the following code:

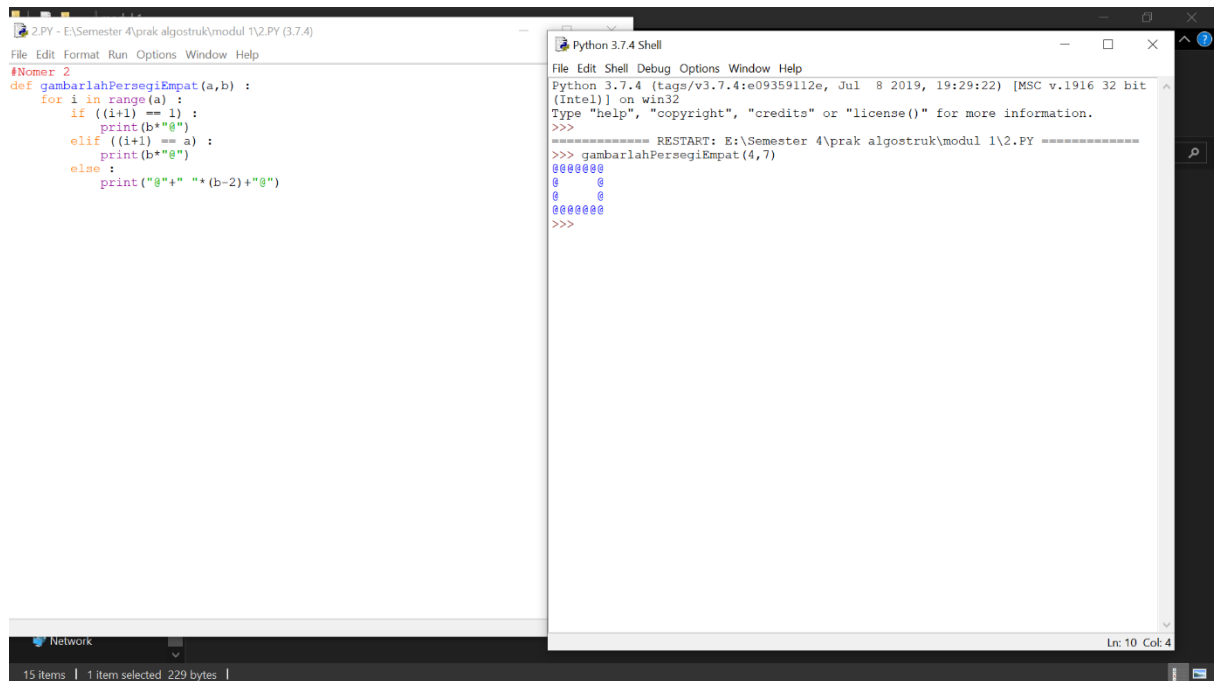
```
#Nomer 1
def cetakSiku(x) :
    for a in range(x) :
        print((a+1)*"****")
```

The right window, titled 'Python 3.7.4 Shell', shows the output of the program after running the function `cetakSiku(6)`:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\1.PY =====
>>> cetakSiku(6)
*
****
*****
*****
*****
*****
>>> |
```

The status bar at the bottom indicates 'Ln: 12 Col: 4'.

2.



The screenshot shows a Python 3.7.4 IDE with two windows. The left window, titled '2.PY - E:\Semester 4\prak algostruk\modul 1\2.PY (3.7.4)', contains the following code:

```
#Nomer 2
def gambarkahPersegiEmpat(a,b) :
    for i in range(a) :
        if ((i+1) == 1) :
            print(b*"****")
        elif ((i+1) == a) :
            print(b*"****")
        else :
            print(" "*(b-2) + "****")
```

The right window, titled 'Python 3.7.4 Shell', shows the output of the program after running the function `gambarkahPersegiEmpat(4,7)`:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\2.PY =====
>>> gambarkahPersegiEmpat(4,7)
*****
*****
*****
*****
>>>
```

The status bar at the bottom indicates 'Ln: 10 Col: 4'.

3.

```
3.PY - E:\Semester 4\prak algostruk\modul 1\3.PY (3.7.4)
File Edit Format Run Options Window Help

#Nomer 3
def jumlahHurufVokal(ch) :
    b = len(ch)
    a = 0
    for i in ch :
        if (i=='A' or i=='a' or i=='E' or i=='e' or i=='I' or i=='i' or i=='O' or i=='o') :
            a += 1
    return b,a

def jumlahHurufKonsonan(ch) :
    b = len(ch)
    a = 0
    for i in ch :
        if (i=='A' or i=='a' or i=='E' or i=='e' or i=='I' or i=='i' or i=='O' or i=='o') :
            a += 1
    return b,b-a

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help

Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\3.PY =====
>>> jumlahHurufVokal('karanganyar')
(11, 4)
>>> jumlahHurufKonsonan('karanganyar')
(11, 7)
>>>
```

4.

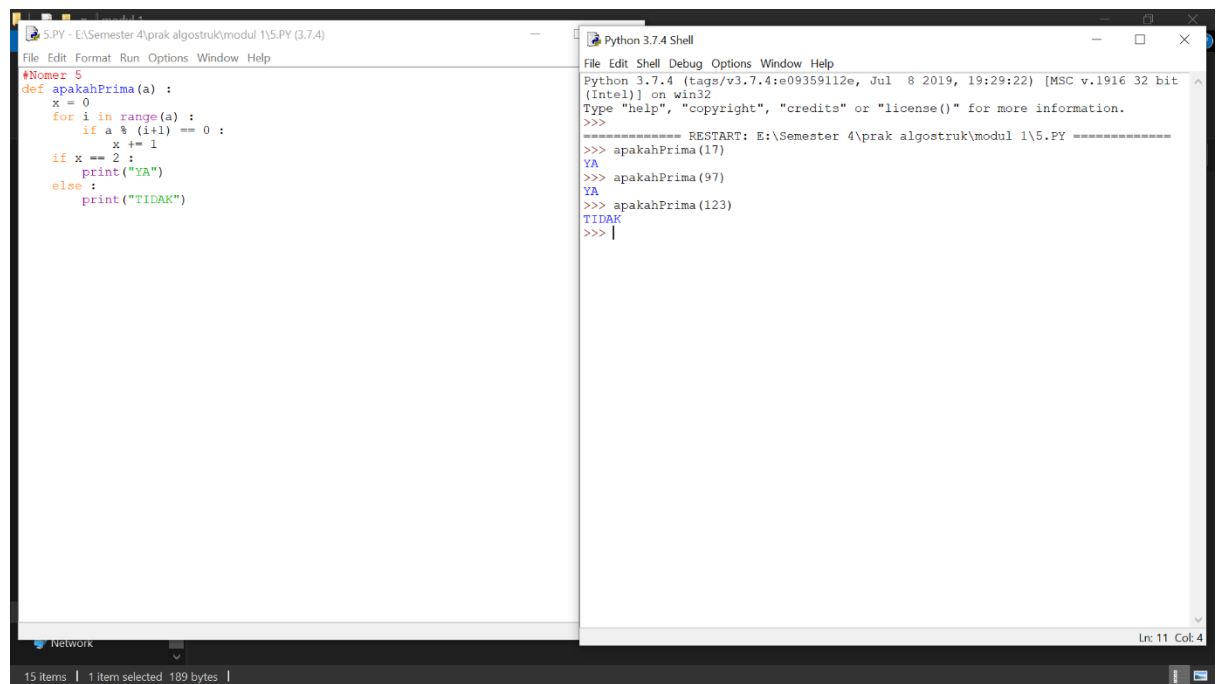
```
4.PY - E:\Semester 4\prak algostruk\modul 1\4.PY (3.7.4)
File Edit Format Run Options Window Help

#Nomer 4
def rerata(x) :
    a = 0
    b = 0
    for i in x :
        a += 1
        b = b + i
    a = float(a)
    b = float(b)
    return (b/a)

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help

Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\4.PY =====
>>> rerata([1,2,3,4,5])
3.0
>>> |
```

5.



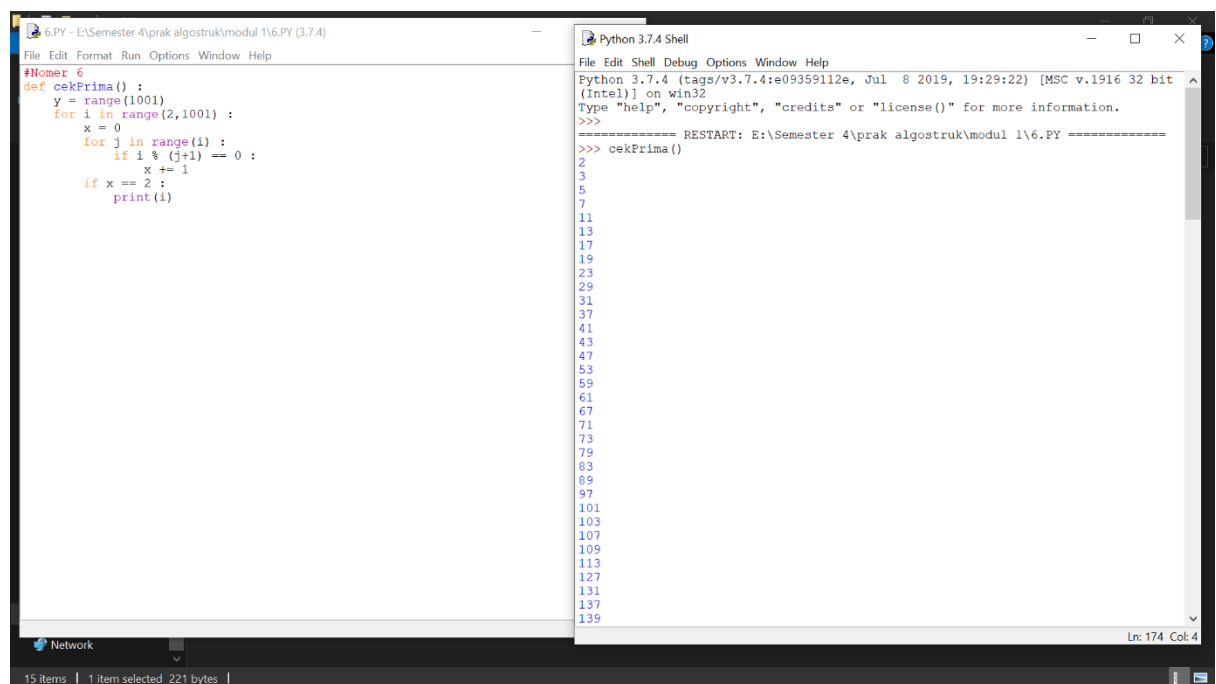
The screenshot shows a Python IDE with a file named '5.PY' and a Python 3.7.4 Shell window. The script in '5.PY' is as follows:

```
#Nomer 5
def apakahPrima(a) :
    x = 0
    for i in range(a) :
        if a % (i+1) == 0 :
            x += 1
    if x == 2 :
        print("YA")
    else :
        print("TIDAK")
```

The shell window shows the execution of the script with the following output:

```
Python 3.7.4 Shell
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\5.PY =====
>>> apakahPrima(17)
YA
>>> apakahPrima(97)
YA
>>> apakahPrima(123)
TIDAK
>>> |
```

6.



The screenshot shows a Python IDE with a file named '6.PY' and a Python 3.7.4 Shell window. The script in '6.PY' is as follows:

```
#Nomer 6
def cekPrima() :
    y = range(1001)
    for i in range(2,1001) :
        x = 0
        for j in range(i) :
            if i % (j+1) == 0 :
                x += 1
        if x == 2 :
            print(i)
```

The shell window shows the execution of the script with the following output:

```
Python 3.7.4 Shell
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\6.PY =====
>>> cekPrima()
2
3
5
7
11
13
17
19
23
29
31
37
41
43
47
53
59
61
67
71
73
79
83
89
97
101
103
107
109
113
127
131
137
139
```

7.

The screenshot shows a Python IDE with two windows. The left window, titled '7.PY - E:\Semester 4\prak algostruk\modul 1\7.PY (3.7.4)', contains the following code:

```
#Nomer 7
def faktorPrima(x):
    listprima=[]
    prima=2
    while prima<=x:
        if x%prima==0:
            x/=prima
            listprima.append(prima)
        else:
            prima+=1
    return listprima
```

The right window, titled 'Python 3.7.4 Shell', shows the execution of the function:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\7.PY =====
>>> faktorPrima(10)
[2, 5]
>>> faktorPrima(120)
[2, 2, 2, 3, 5]
>>> faktorPrima(19)
[19]
>>> |
```

8.

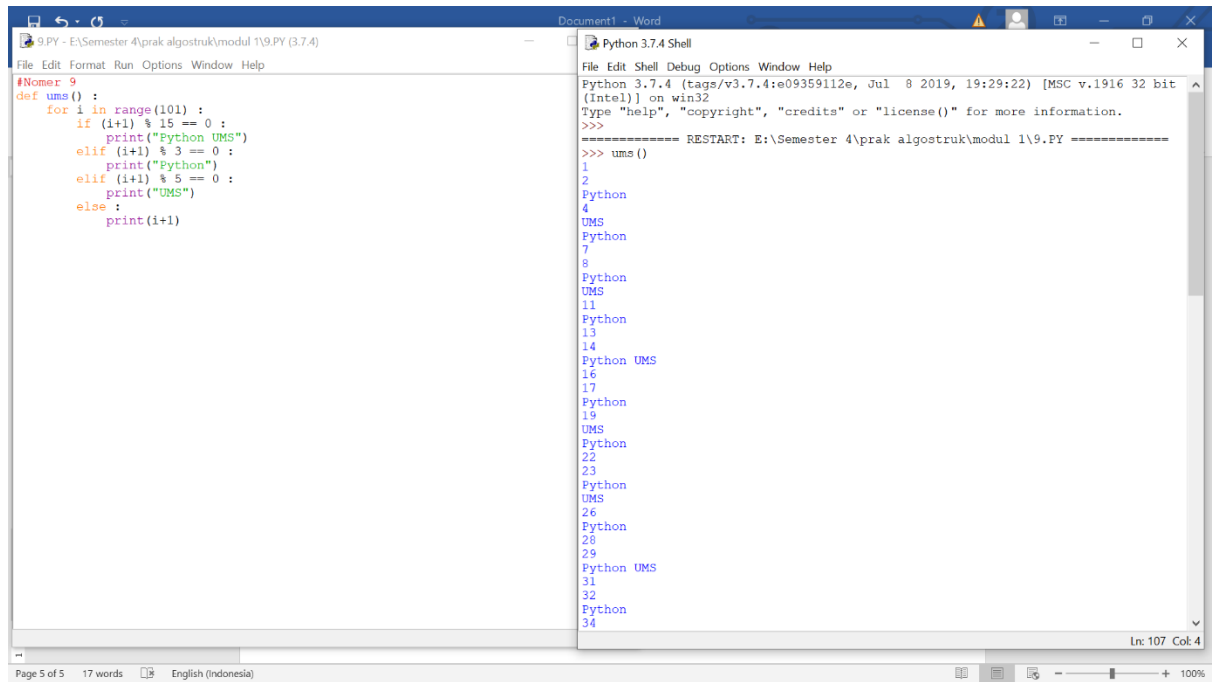
The screenshot shows a Python IDE with two windows. The left window, titled '8.PY - E:\Semester 4\prak algostruk\modul 1\8.PY (3.7.4)', contains the following code:

```
#Nomer 8
def apakahTerkandung(a,b) :
    if a in b :
        return True
    else :
        return False
```

The right window, titled 'Python 3.7.4 Shell', shows the execution of the function:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\8.PY =====
>>> h = 'do'
>>> k = 'Indonesia tanah air beta'
>>> apakahTerkandung('pusaka', k)
False
>>>
```

9.



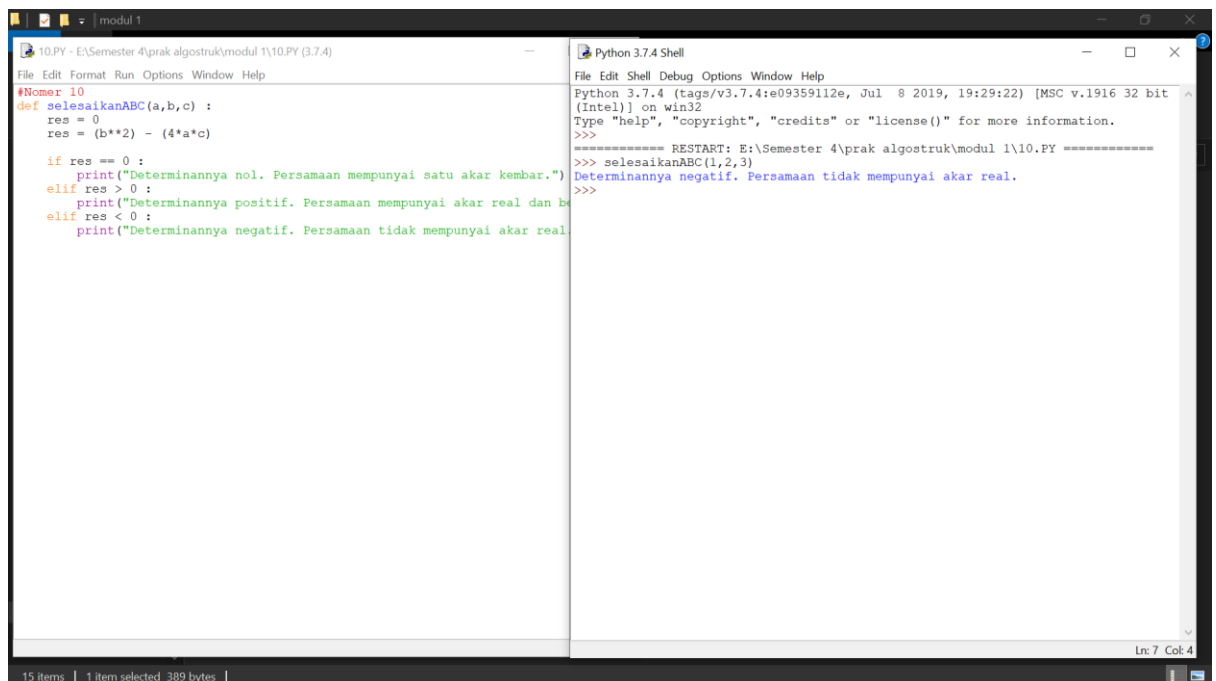
The screenshot shows a Python IDE with two windows. The left window, titled '9.PY - E:\Semester 4\prak algostruk\modul 1\9.PY (3.7.4)', contains the following code:

```
#Nomer 9
def ums():
    for i in range(101):
        if (i+1) % 15 == 0:
            print("Python UMS")
        elif (i+1) % 3 == 0:
            print("Python")
        elif (i+1) % 5 == 0:
            print("UMS")
        else:
            print(i+1)
```

The right window, titled 'Python 3.7.4 Shell', shows the output of the program:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\9.PY =====
>>> ums()
1
2
Python
4
UMS
Python
7
8
Python
UMS
11
Python
13
14
Python UMS
16
17
Python
19
UMS
Python
22
23
Python
UMS
26
Python
28
29
Python UMS
31
32
Python
34
```

10.



The screenshot shows a Python IDE with two windows. The left window, titled '10.PY - E:\Semester 4\prak algostruk\modul 1\10.PY (3.7.4)', contains the following code:

```
#Nomer 10
def selesaikanABC(a,b,c):
    res = 0
    res = (b**2) - (4*a*c)

    if res == 0:
        print("Determinannya nol. Persamaan mempunyai satu akar kembar.")
    elif res > 0:
        print("Determinannya positif. Persamaan mempunyai akar real dan berbeda.")
    elif res < 0:
        print("Determinannya negatif. Persamaan tidak mempunyai akar real.")
```

The right window, titled 'Python 3.7.4 Shell', shows the output of the program:

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\10.PY =====
>>> selesaikanABC(1,2,3)
Determinannya negatif. Persamaan tidak mempunyai akar real.
>>>
```

11.

The screenshot shows a Python 3.7.4 IDE with a file named 11.PY. The script defines a function `apakahKabisat()` that checks if a year is a leap year. The shell shows the execution of the script, which prompts the user to enter a year (2020) and outputs `True`.

```

11.PY - E:\Semester 4\prak algostruk\modul 1\11.PY (3.7.4)
File Edit Format Run Options Window Help
#Nomer 11
def apakahKabisat() :
    thn = int(input("Masukkan Tahun : "))
    if thn % 4 == 0 :
        if thn % 100 == 0 :
            if thn % 400 == 0 :
                print ('True')
            else :
                print ('False')
        else :
            print ('True')
    else :
        print ('False')

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\11.PY =====
>>> apakahKabisat()
Masukkan Tahun : 2020
True
>>>

```

12.

The screenshot shows a Python 3.7.4 IDE with a file named 12.PY. The script defines a function `tebak()` that plays a number guessing game. The shell shows the execution of the script, which prompts the user to enter a number (50) and outputs `Ya. Anda benar.`

```

12.PY - E:\Semester 4\prak algostruk\modul 1\12.PY (3.7.4)
File Edit Format Run Options Window Help
#Nomer 12
import random
def tebak() :
    a = random.randrange(1,101)
    b = -1
    n = 0
    print("Permainan tebak angkat.")
    print("Saya menyimpan sebuah angka bulat antara 1 sampai 100. Coba tebak")
    while a != b :
        n = n + 1
        b = int(input("Masukkan tebakan ke-"+str(n)+"> "))
        if b < a :
            print("Itu terlalu kecil. Coba lagi")
        elif b > a :
            print("Itu terlalu besar. Coba lagi")
        else :
            print("Ya. Anda benar.")
            break

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\12.PY =====
>>> tebak()
Traceback (most recent call last):
  File "<pyshell#0>", line 1, in <module>
    tebak()
  File "E:\Semester 4\prak algostruk\modul 1\12.PY", line 3, in tebak
    a = random.randrange(1,101)
NameError: name 'random' is not defined
>>> tebak()
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in <module>
    tebak()
  File "E:\Semester 4\prak algostruk\modul 1\12.PY", line 3, in tebak
    a = random.randrange(1,101)
NameError: name 'random' is not defined
>>>
===== RESTART: E:\Semester 4\prak algostruk\modul 1\12.PY =====
>>> tebak()
Permainan tebak angkat.
Saya menyimpan sebuah angka bulat antara 1 sampai 100. Coba tebak
Masukkan tebakan ke-1:> 50
Itu terlalu kecil. Coba lagi
Masukkan tebakan ke-2:> 75
Itu terlalu besar. Coba lagi
Masukkan tebakan ke-3:> 58
Itu terlalu kecil. Coba lagi
Masukkan tebakan ke-4:> 60
Itu terlalu kecil. Coba lagi
Masukkan tebakan ke-5:> 65
Itu terlalu kecil. Coba lagi
Masukkan tebakan ke-6:> 70
Itu terlalu kecil. Coba lagi
Masukkan tebakan ke-7:> 73
Itu terlalu kecil. Coba lagi
Masukkan tebakan ke-8:> 74
Ya. Anda benar.
>>>

```

13.

The screenshot shows a Python IDE with a file named '13.PY'. The script defines a function `katakan(x)` that converts a number `x` into its Indonesian word representation. The function uses a list `satuan` for units and a series of `elif` statements to handle tens, hundreds, thousands, and larger units like millions and billions. The script then calls `katakan(3234500)` and prints the result.

```
#Nomer 13
def katakan(x):
    satuan = [' ', 'satu', 'dua', 'tiga', 'empat', 'lima', 'enam', 'tujuh', 'delapan', 'sembilan', 'sepuluh', 'satu puluh']
    hasil = ""
    if x <= 0:
        hasil += 'Bilangan Haruslah Positif\ndan Bilangan Asli'
    elif x < 12:
        hasil += satuan[x]
    elif x < 20:
        hasil += katakan(x-10) + " belas "
    elif x < 100:
        hasil += katakan(int(x/10)) + " puluh " + katakan(x%10)
    elif x < 200:
        hasil += "seratus " + katakan(x-100)
    elif x < 1000:
        hasil += katakan(int(x/100)) + " ratus " + katakan(x%100)
    elif x < 2000:
        hasil += "seribu " + katakan(x-1000)
    elif x < 1000000:
        hasil += katakan(int(x/1000)) + " ribu " + katakan(x%1000)
    elif x < 1000000000:
        hasil += katakan(int(x/1000000)) + " juta " + katakan(x%1000000)
    elif x >= 1000000000:
        hasil += katakan(int(x/1000000000)) + " milyar " + katakan(x%1000000000)
    return hasil

>>> katakan(3234500)
'tiga juta dua ratus tiga puluh empat ribu lima ratus Bilangan Haruslah Positif\ndan Bilangan Asli'
>>>
```

The Python Shell shows the execution of the script, displaying the output of the `katakan` function call.

14.

The screenshot shows a Python IDE with a file named '14.PY'. The script defines a function `formatRupiah(a)` that takes a number `a` and formats it as a string with commas as thousands separators. The function uses a loop to insert commas at every third digit from the right. The script then calls `formatRupiah(1500)` and `formatRupiah(2560000)` and prints the results.

```
#Nomer 14
def formatRupiah(a):
    a = list(str(a))
    b = len(a)
    if b % 3 == 0:
        b = int(b/3) - 1
    else:
        b = int(b/3)
    n = 0
    for i in range(b):
        x = -3*(i+1)
        a.insert(int(x)+n, ",")
        n = n - 1
    a = "".join(a)
    print("Rp "+a)

>>> formatRupiah(1500)
Rp 1.500
>>> formatRupiah(2560000)
Rp 2.560.000
>>>
```

The Python Shell shows the execution of the script, displaying the output of the `formatRupiah` function calls.

