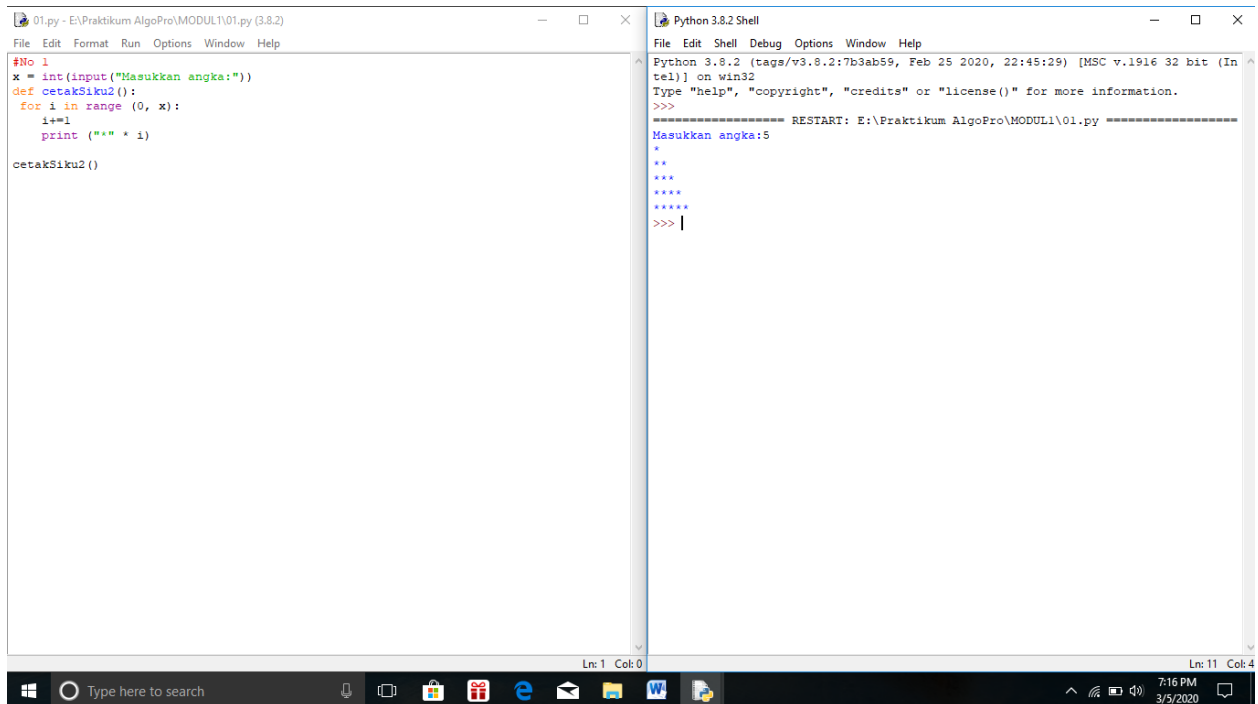


Nama : Willi Susanti

NIM : L200180060

Laporan Praktikum Algoritma dan Struktur Data

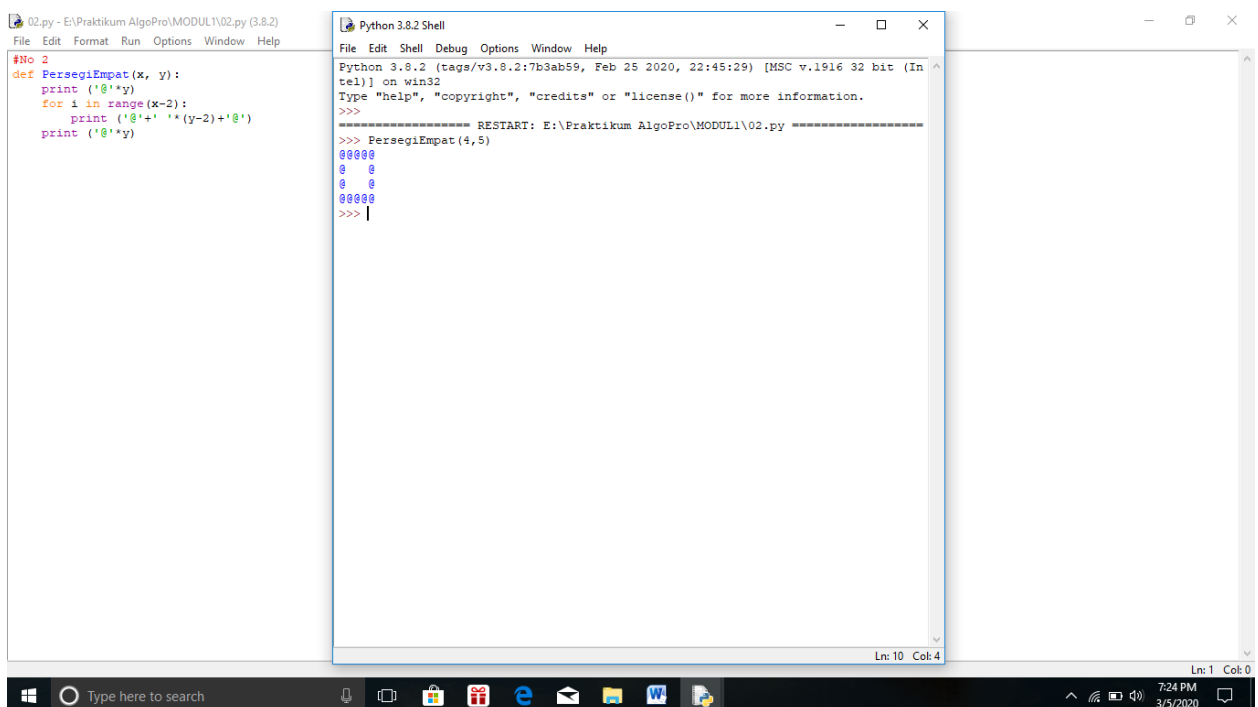
1.



```
#No 1
x = int(input("Masukkan angka:"))
def cetakSiku2():
    for i in range(0, x):
        i+=1
        print ("*" * i)
cetakSiku2()
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\01.py =====
Masukkan angka:5
*
**
***
****
*****
>>> |
```

2.



```
#No 2
def PersegiEmpat(x, y):
    print ('@'*y)
    for i in range(x-2):
        print ('@'+ ' '* (y-2) + '@')
    print ('@'*y)
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\02.py =====
>>> PersegiEmpat(4,5)
@@@@@
@  @
@  @
@@@@@
>>> |
```

3.a

The screenshot shows a Python IDE with two windows. The left window, titled '03a.py - E:\Praktikum AlgoPro\MODUL1\03a.py (3.8.2)', contains the following code:

```
#No 3 a
def jumlahHurufVokal(x):
    vokal = "AIUEOaiueo"
    a = len(x)
    b = ""
    for k in x:
        if k in vokal:
            b += k
    c = len(b)
    return (a, c)
```

The right window, titled 'Python 3.8.2 Shell', shows the execution of the script. It displays the restart message and the output for the function call `jumlahHurufVokal('Surakarta')`, which returns `(9, 4)`.

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\03a.py =====
>>> k = jumlahHurufVokal ('Surakarta')
>>> k
(9, 4)
>>>
```

The taskbar at the bottom shows the Windows search bar, taskbar icons, and system clock indicating 7:33 PM on 3/5/2020.

3.b

The screenshot shows a Python IDE with two windows. The left window, titled '03b.py - E:\Praktikum AlgoPro\MODUL1\03b.py (3.8.2)', contains the following code:

```
#No 3 b
def jumlahHurufKonsonan(x):
    konsonan = "BCDFGHJKLMNPQRSTVWXYZbcdfghjklmnpqrstvwxyz"
    a = len(x)
    b = ""
    for k in x:
        if k in konsonan:
            b += k
    c = len(b)
    return (a, c)
```

The right window, titled 'Python 3.8.2 Shell', shows the execution of the script. It displays the restart message and the output for the function call `jumlahHurufKonsonan('Surakarta')`, which returns `(9, 5)`.

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\03b.py =====
>>> k = jumlahHurufKonsonan('Surakarta')
>>> k
(9, 5)
>>>
```

The taskbar at the bottom shows the Windows search bar, taskbar icons, and system clock indicating 7:36 PM on 3/5/2020.

4.

```
#No 4
def rerata(b):
    k = 0
    for i in (b):
        k+=1
    hasil = k / len(b)
    return hasil
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\04.py =====
>>> rerata([1,2,3,4,5])
3.0
>>> g = [3,4,5,4,3,4,5,2,2,10,11,23]
>>> rerata(g)
6.333333333333333
>>>
```

5.

```
#No 5
from math import sqrt as sq
def apakahPrima(x):
    x = int(x)
    primaKecil = [2,3,5,7,11]
    bknPrimaKecil = [0,1,4,6,8,9,10]
    if x in primaKecil:
        return True
    elif x in bknPrimaKecil:
        return False
    else:
        for i in range(2, int(sq(x))+1):
            if x % i == 0:
                return False
            else:
                return True
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\05.py =====
>>> apakahPrima(17)
True
>>> apakahPrima(97)
True
>>> apakahPrima(123)
False
>>> |
```

6.

```
06.py - E:\Praktikum AlgoPro\MODUL1\06.py (3.8.2)
File Edit Format Run Options Window Help

#No 6
def apePrima(x,y):
    lower = 2
    upper = 1000
    print("Bilangan prima antara",lower,"and",upper,":")
    for num in range(lower,upper + 1):
        if num > 1:
            for i in range(2,num):
                if (num % i) == 0:
                    break
            else:
                print(num)
                break

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help

923
925
927
929
931
933
935
937
939
941
943
945
947
949
951
953
955
957
959
961
963
965
967
969
971
973
975
977
979
981
983
985
987
989
991
993
995
997
999
>>>
```

7.

```
07.py - E:\Praktikum AlgoPro\MODUL1\07.py (3.8.2)
File Edit Format Run Options Window Help

#No 7
def faktorprima(x):
    faktor=[]
    loop=2
    while loop<=x:
        if x%loop==0:
            x/=loop
            faktor.append(loop)
        else:
            loop+=1
    return faktor

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\07.py =====
>>> faktorprima(10)
[2, 5]
>>> faktorprima(120)
[2, 2, 2, 3, 5]
>>> faktorprima(19)
[19]
>>>
```

8.

The screenshot shows a Python 3.8.2 IDE with a file named '08.py'. The code defines a function `apakahTerkandung(x, y)` that checks if `x` is a substring of `y`. The function returns `True` if `x` is in `y`, and `False` otherwise. The shell window shows the execution of the function with `h='do'` and `k='indonesia tanah air beta'`, returning `True`, and with `k='pusaka'`, returning `False`.

```
#No 8
def apakahTerkandung(x, y):
    for k in x:
        if k in y:
            return True
        else:
            return False

>>> h='do'
>>> k='indonesia tanah air beta'
>>> apakahTerkandung(h,k)
True
>>> apakahTerkandung('pusaka',k)
False
>>> |
```

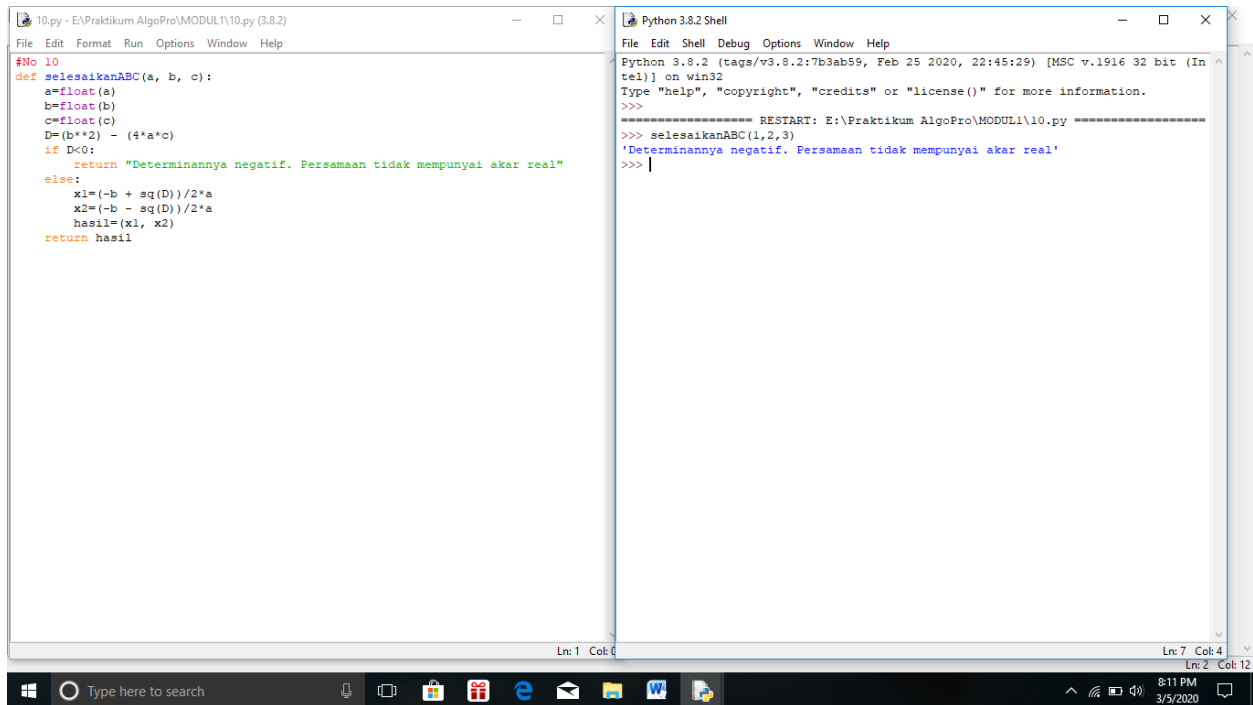
9.

The screenshot shows a Python 3.8.2 IDE with a file named '09.py'. The code defines a function `rubah35(a, b)` that prints the number of times 'Python' and 'UMS' appear in a string `s` between indices `a` and `b`. The shell window shows the execution of the function with `a=1` and `b=100`, printing the results for each index from 1 to 100.

```
#No 9
def rubah35(a,b):
    a = 1
    b = 100
    for i in range(a, b+1):
        if (i % 3) == 0 and (i % 5) == 0:
            print("Python UMS")
        elif (i % 3) == 0:
            print("Python")
        elif (i % 5) == 0:
            print("UMS")
        else:
            print(i)

>>> |
```

10.

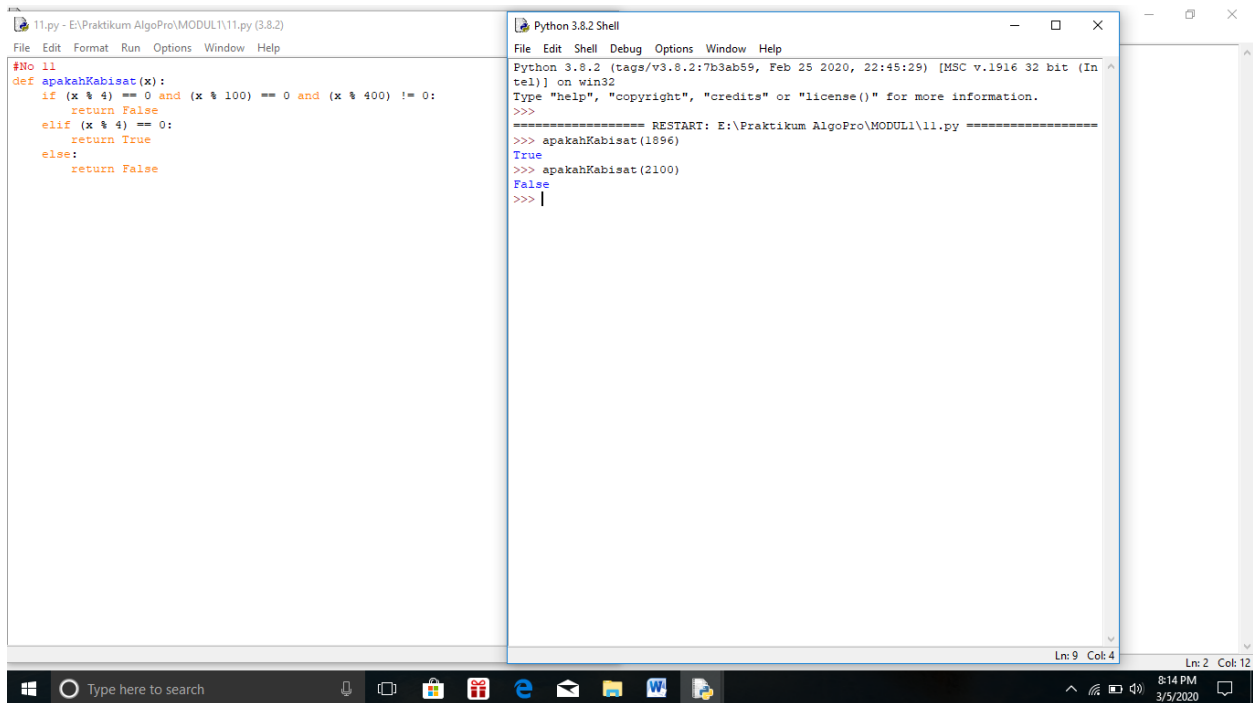


The screenshot shows a Python IDE with two windows. The left window is a text editor for a file named `10.py` located at `E:\Praktikum AlgoPro\MODUL1\10.py`. It contains a function `selesaikanABC(a, b, c)` that calculates the roots of a quadratic equation $ax^2 + bx + c = 0$. The function uses `float` for coefficients, calculates the discriminant $D = b^2 - 4ac$, and returns the roots x_1 and x_2 if $D \geq 0$, or a message 'Determinannya negatif. Persamaan tidak mempunyai akar real' if $D < 0$. The right window is a 'Python 3.8.2 Shell' showing the execution of the function with arguments `(1, 2, 3)`, which results in the same negative discriminant message. The Windows taskbar at the bottom shows the time as 8:11 PM on 3/5/2020.

```
#No 10
def selesaikanABC(a, b, c):
    a=float(a)
    b=float(b)
    c=float(c)
    D=(b**2) - (4*a*c)
    if D<0:
        return "Determinannya negatif. Persamaan tidak mempunyai akar real"
    else:
        x1=(-b + sq(D))/2*a
        x2=(-b - sq(D))/2*a
        hasil=(x1, x2)
    return hasil
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\10.py =====
>>> selesaikanABC(1,2,3)
'Determinannya negatif. Persamaan tidak mempunyai akar real'
>>> |
```

11.



The screenshot shows a Python IDE with two windows. The left window is a text editor for a file named `11.py` located at `E:\Praktikum AlgoPro\MODUL1\11.py`. It contains a function `apakahKabisat(x)` that checks if a year `x` is a leap year (kabisat) based on the rules: divisible by 4 but not by 100, or divisible by 400. The function returns `True` for kabisat years and `False` otherwise. The right window is a 'Python 3.8.2 Shell' showing the execution of the function with arguments `(1896)` and `(2100)`, returning `True` and `False` respectively. The Windows taskbar at the bottom shows the time as 8:14 PM on 3/5/2020.

```
#No 11
def apakahKabisat(x):
    if (x % 4) == 0 and (x % 100) == 0 and (x % 400) != 0:
        return False
    elif (x % 4) == 0:
        return True
    else:
        return False
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\11.py =====
>>> apakahKabisat(1896)
True
>>> apakahKabisat(2100)
False
>>> |
```

12.

The screenshot shows a Python 3.8.2 IDE with two windows. The left window, titled '12.py - E:\Praktikum AlgoPro\MODUL1\12.py (3.8.2)', contains a script for a number guessing game. The script generates a random number between 1 and 100 and allows the user to guess it up to three times. The right window, titled 'Python 3.8.2 Shell', shows the execution of the script. The user has entered three guesses: 50, 75, and 58. The script correctly identifies the number as 13 after the third guess.

```
#No 12
import random

print("""Permainan tebak angka.
Saya menyimpan sebuah angka bulat antara 1 sampai 100. Coba Tebak!""")
a = random.randint(1,100)
for i in range (3):
    b = int(input("Masukkan tebakkan ke-{}:>".format(i+1)))
    if b == a:
        print ("Ya. Anda benar.")
    elif b > a:
        if i >= 2:
            print ("Itu terlalu besar. Kesempatan habis. Nilainya adalah",a)
        else:
            print ("Itu terlalu besar. Coba lagi")
    else:
        if i >= 2:
            print ("Itu terlalu kecil. Kesempatan habis. Nilainya adalah",a)
        else:
            print ("Itu terlalu kecil. Coba lagi")
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\12.py =====
Permainan tebak angka.
Saya menyimpan sebuah angka bulat antara 1 sampai 100. Coba Tebak!
Masukkan tebakkan ke-1:>50
Itu terlalu besar. Coba lagi
Masukkan tebakkan ke-2:>75
Itu terlalu besar. Coba lagi
Masukkan tebakkan ke-3:>58
Itu terlalu besar. Kesempatan habis. Nilainya adalah 13
>>>
```

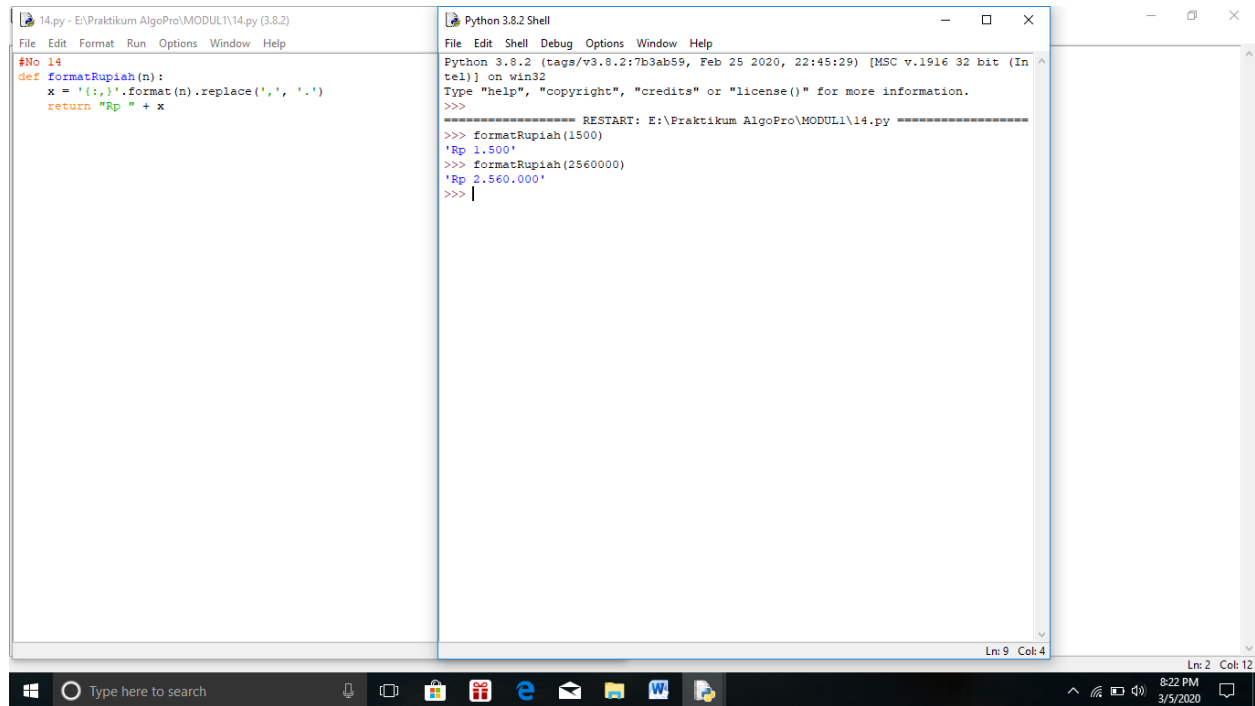
13.

The screenshot shows a Python 3.8.2 IDE with two windows. The left window, titled '13.py - E:\Praktikum AlgoPro\MODUL1\13.py (3.8.2)', contains a script that converts a numerical value into its corresponding word representation in Indonesian. The script uses a list of words for units and recursive calls for larger numbers. The right window, titled 'Python 3.8.2 Shell', shows the execution of the script. The user has entered the number 3125750, and the script outputs 'tiga jutaseratusduapuluhlimaributujuh ratuslimapuluh'.

```
#No 13
def katakan(x):
    satuan = ['', 'satu', 'dua', 'tiga', 'empat', 'lima', 'enam', 'tujuh', 'delapan', 'sembilan', 'sepuluh', 'sebelas']
    hasil = ""
    if x <= 0:
        hasil += ""
    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
```

```
Python 3.8.2 Shell
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\13.py =====
>>> katakan(3125750)
'tiga jutaseratusduapuluhlimaributujuh ratuslimapuluh'
>>>
```

14.



The image shows a screenshot of a Python IDE with two windows. The left window, titled '14.py - E:\Praktikum AlgoPro\MODUL1\14.py (3.8.2)', contains the following code:

```
#No 14
def formatRupiah(n):
    x = '{:,}'.format(n).replace(', ', '.')
    return "Rp " + x
```

The right window, titled 'Python 3.8.2 Shell', shows the execution of the script. It displays the Python version and architecture, followed by a restart command and the execution of the `formatRupiah` function with three test values:

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\Praktikum AlgoPro\MODUL1\14.py =====
>>> formatRupiah(1500)
'Rp 1.500'
>>> formatRupiah(2560000)
'Rp 2.560.000'
>>>
```

The taskbar at the bottom shows the Windows search bar, taskbar icons, and system clock indicating 8:22 PM on 3/5/2020.