

Tugas Pemrograman Section 3

Ahmad Agung Tawkkal

1184015

D4 TI 1B

October 2019

1 Teori

1.1 Fungsi

Fungsi adalah suatu balok program yang terdiri dari nama fungsi, input variable dan variable kembalian. Nama fungsi diawali dengan *def* dan setelahnya tanda titik dua. Nama bisa sama dengan isi berbeda jika menggunakan huruf besar dan kecil atau sering disebut dengan *case sensitive*. Input variable bisa lebih dari satu dengan pemisahan tanda koma. Variable kembalian pasti satu, bebas apakah itu jenis *string*, *integer*, *list* atau *dictionary*. Contohnya sebagai berikut:

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 19:38:13 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8
9 def salam():
10     nama = input()
11     print ("Hello , Selamat Pagi "+nama)
12
13 salam()
14
15
16 #Output
17 #Agung                -inputan
18 #Hello , Selamat Pagi Agung -hasil
```

1.2 Library

Library digunakan untuk memanggil fungsi pada file python yang berbeda, library bertujuan untuk memanggil fungsi agar dapat memisahkan program di setiap fungsi.

Pada file dibawah terdapat ada dua fungsi yaitu *salam* dan *nama lengkap*

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 20:20:51 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 def salam():
9     nama = input()
10     print ("Hello , Selamat Pagi "+nama)
11
12 salam()
13
14 def nama_lengkap():
15     depan=input()
```

```

16     belakang=input()
17     print ("Hay "+depan+belakang)
18
19 nama_lengkap()

```

Kemudian untuk memanggil fungsinya anda tinggal mengimport.

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Wed Oct 30 20:16:55 2019
4
5  @author: Ahmad Agung Tawakkal
6  """
7
8  #contoh menggunakan import
9
10 import fungsi
11
12 c=fungsi.salam()
13 print (c)

```

1.3 Kelas, Objek, Atribut, dan Method

1.3.1 Kelas

Kelas menyediakan sarana untuk menggabungkan data dan fungsionalitas bersama. Membuat kelas baru menciptakan tipe objek baru, memungkinkan instance baru dari tipe itu dibuat. Setiap instance kelas dapat memiliki atribut yang melekat padanya untuk mempertahankan negaranya. Instance kelas juga dapat memiliki metode (ditentukan oleh kelasnya) untuk memodifikasi kondisinya.

1.3.2 Objek

Pemrograman berorientasi objek atau dalam bahasa inggris disebut *Object Oriented Programming* (OOP) adalah paradigma atau teknik pemrograman di mana semua hal dalam program dimodelkan seperti objek dalam dunia nyata. Objek di dunia nyata memiliki ciri atau atribut dan juga aksi atau kelakuan (*behaviour*).

1.3.3 Atribut

Atribut adalah nilai data yang terdapat pada suatu object yang berasal dari class. Attributes merepresentasikan karakteristik dari suatu object.

1.3.4 Method

Method adalah fungsi atau prosedur yang dibuat oleh seorang programmer didalam suatu Class. Method dapat dibagi menjadi fungsi dan prosedur. Fungsi adalah bagian atau sub dari program yang mempunyai algoritma tertentu dalam

menyelesaikan suatu masalah dengan mengembalikan hasil. Prosedur adalah bagian atau sub dari program yang mempunyai algoritma tertentu dalam menyelesaikan suatu masalah tanpa mengembalikan suatu nilai hasil.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 20:46:08 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8
9 class Bapak(object):
10     def __init__(self, nama, tinggi, berat):
11         self.nama = nama
12         self.tinggi = tinggi
13         self.berat = berat
14
15     def berjalan(self):
16         print("Berjalan ke depan")
17
18     def berlari(self):
19         print("Berlari dengan cepat")
20
21
22 # class Anak turunan dari class Bapak
23 class Anak(Bapak):
24     pass
25
26
27 b = Bapak("Wiragan", 170, 68)
28 print()
29 print("Nama:", b.nama)
30 print("Tinggi:", b.tinggi, "cm")
31 print("Berat:", b.berat, "kg")
32 b.berjalan()
33 b.berlari()
34
35 # objek dari class Anak memiliki seluruh yang dimiliki class Bapak
36 a = Anak("Mustofa", 140, 32)
37 print()
38 print("Nama:", a.nama)
39 print("Tinggi:", a.tinggi, "cm")
40 print("Berat:", a.berat, "kg")
41 a.berjalan()
42 a.berlari()
```

1.4 Penggunaan Library

Contoh file yang akan dipanggil dengan from :

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 20:20:51 2019
4
5 @author: Ahmad Agung Tawakkal
```

```

6 """
7
8 def salam():
9     nama = input()
10    print ("Hello , Selamat Pagi "+nama)
11
12 salam()
13
14 def nama_lengkap():
15     depan=input()
16     belakang=input()
17     print ("Hay "+depan+belakang)
18
19 nama_lengkap()

```

Contoh pemanggilan library:

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 20:16:55 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 #Contoh menggunakan from
9
10 from fungsi import salam
11
12 c=salam()
13 print (c)

```

1.5 Import Kalkulator

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 22:16:58 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 def Penambahan(a,b):
9     r = a + b
10    return r
11 def Pengurangan(a,b):
12     r = a - b
13    return r
14 def Perkalian(a,b):
15     r = a * b
16    return r
17 def Pembagian(a,b):
18     r = a / b
19    return r

```

```

1 # -*- coding: utf-8 -*-
2 """

```

```
3 Created on Wed Oct 30 23:03:32 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 from kalkulator import Penambahan #memanggil file kalkulator dari
9     library kemudian memanggil fungsi penambahan
10
11 c=Penambahan(3,2)
12 print (c)
```

Output = 5

1.6 Pemanggilan library dalam sebuah folder

Untuk mengakses sebuah library dalam folder, terlebih dahulu foldernya kita tulis(src) kemudian mengimport nama librarynya(soal1).

Contoh: `from src import libra`

1.7 Pemanggilan class dalam sebuah folder

Untuk mengakses sebuah class dalam sebuah folder, terlebih dahulu menuliskan foldernya, kemudian mengimport nama class-nya.

contoh: `from src import Buah`

2 Keterampilan Pemrograman

Soal 1

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 19:38:13 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 def NPM1(npm):
9     print(" *** **          ****          *** **      *****      ***      *****
10 ")
11     print(" *** **      ***          ***      ***      ***      ***      ***      ***      *****
12 ")
13     print(" *** **      ***          ***      ***      ***      ***      ***      ***      ***
14 ")
15     print(" *** **          ****          *****      ***      ***      ***      ***      *****
16 ")
17     print(" *** **          ****          *****      ***      ***      ***      ***      *****
18 ")
19     print(" *** **      ***          ***          ***          ***      ***      ***      ***      *****
20 ")
21     print(" *** **      ***          ***          ***          ***          ***      ***      ***      *****
22 ")
23     print(" *** **      ***          ***          ***          ***          ***          ***      ***      *****
24 ")
25     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
26 ")
27     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
28 ")
29     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
30 ")
31     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
32 ")
33     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
34 ")
35     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
36 ")
37     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
38 ")
39     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
40 ")
41     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
42 ")
43     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
44 ")
45     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
46 ")
47     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
48 ")
49     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
50 ")
51     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
52 ")
53     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
54 ")
55     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
56 ")
57     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
58 ")
59     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
60 ")
61     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
62 ")
63     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
64 ")
65     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
66 ")
67     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
68 ")
69     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
70 ")
71     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
72 ")
73     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
74 ")
75     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
76 ")
77     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
78 ")
79     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
80 ")
81     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
82 ")
83     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
84 ")
85     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
86 ")
87     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
88 ")
89     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
90 ")
91     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
92 ")
93     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
94 ")
95     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
96 ")
97     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
98 ")
99     print(" *** **      ***          ***          ***          ***          ***          ***          ***      *****
100 ")

```

```
16     print(" *** **          ****          ***          ****          ***          ****")
17     NPM1(npm)
```

Soal 2

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 23:27:12 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 def NPM2(npm) :
9     npm = int (npm)
10     Tld=abs(npm)%100
11     for i in range(Tld):
12         print (" Hallo , ",npm," apa kabar ?")
13
14 NPM2(npm)

```

Soal 3

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 30 23:27:12 2019
4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 def NPM3(npm):
9     for i in range(int (str(npm)[4]) + int (str(npm)[5]) + int (str
10         (npm)[6])):
11         print ("Halo, " + str(npm)[4] + str(npm)[5] + str(npm)[6] +
12             " apa kabar ?")
13
14 i = 0
15 npm = input ("Masukan NPM : ")
16 while i < 1:
17     if len (npm) < 7:
18         print ("NPM Kurang dari 7 digit")
19         npm = input ("Masukan NPM : ")
20     elif len (npm) > 7:
21         print ("NPM lebih dari 7 digit")
22         npm = input ("Masukan NPM : ")
23     else :
24         i = 1
25 NPM3(npm)

```

Soal 4

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Oct 31 00:17:55 2019
```

```

4
5 @author: Ahmad Agung Tawakkal
6 """
7
8 def NPM4(npm):
9     NPM = input("Masukan NPM : ")
10    print("Halo, ",NPM[4], " apa kabar ?")
11 NPM4(npm)

```

Soal 5

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Oct 22 14:16:19 2019
4
5 @author: ahmad agung tawakkal
6 """
7
8 def NPM5(npm):
9
10    i=0
11    NPM = input("NPM : ")
12    while i<1:
13        if len(NPM)<7:
14            print("NPM kurang dari 7!")
15            NPM = input("Npm : ")
16        elif len(NPM)>7:
17            print("NPM lebih dari 7!")
18            NPM = input("NPM11: ")
19        else:
20            i=1
21
22    A=NPM[0]
23    B=NPM[1]
24    C=NPM[2]
25    D=NPM[3]
26    E=NPM[4]
27    F=NPM[5]
28    G=NPM[6]
29
30    for this in A,B,C,D,E,F,G:
31        print(this,end = " ")
32
33 NPM5(npm)

```

Soal 6

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Oct 22 14:16:19 2019
4
5 @author: ahmad agung tawakkal
6 """
7
8 def NPM6(npm):

```



```

9
10 i=0
11 NPM = input("NPM : ")
12 while i < 1:
13     if len(NPM) < 7:
14         print("NPM kurang dari 7!")
15         NPM = input("NPM : ")
16     elif len(NPM) > 7:
17         print("NPM lebih dari 7!")
18         NPM = input("NPM : ")
19     else:
20         i=1
21
22 A=NPM[0]
23 B=NPM[1]
24 C=NPM[2]
25 D=NPM[3]
26 E=NPM[4]
27 F=NPM[5]
28 G=NPM[6]
29
30 X=0
31
32 for this in A,B,C,D,E,F,G:
33     X+=int(this)
34     print(X)
35
36 NPM6(npm)

```

Soal 7

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Oct 22 14:16:19 2019
4
5 @author: ahmad agung tawakkal
6 """
7
8 def NPM7(npm):
9
10     i=0
11     NPM = input("NPM : ")
12     while i < 1:
13         if len(NPM) < 7:
14             print("NPM kurang dari 7!")
15             NPM = input("NPM : ")
16         elif len(NPM) > 7:
17             print("NPM lebih dari 7!")
18             NPM = input("NPM : ")
19         else:
20             i=1
21
22     A=NPM[0]
23     B=NPM[1]
24     C=NPM[2]
25     D=NPM[3]
26     E=NPM[4]

```

```

27 F=NPM[5]
28 G=NPM[6]
29
30 X=1
31
32 for this in A,B,C,D,E,F,G:
33     X*=int(this)
34     print(X)
35
36 NPM7(npm)

```

Soal 8

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Oct 22 14:16:19 2019
4
5  @author: ahmad agung tawakkal
6  """
7
8  def NPM8(npm) :
9
10     i=0
11     NPM = input("NPM : ")
12     while i<1:
13         if len(NPM)<7:
14             print("NPM kurang dari 7!")
15             NPM = input("NPM : ")
16         elif len(NPM)>7:
17             print("NPM lebih dari 7!")
18             NPM = input("NPM : ")
19         else :
20             i=1
21
22     A=NPM[0]
23     B=NPM[1]
24     C=NPM[2]
25     D=NPM[3]
26     E=NPM[4]
27     F=NPM[5]
28     G=NPM[6]
29
30     X=1
31
32     for this in A,B,C,D,E,F,G:
33
34         if int(this)%2==0:
35             if int(this)==0:
36                 this=""
37             print(this,end = " ")
38
39     NPM8(npm)

```

Soal 9

```

1  # -*- coding: utf-8 -*-

```

```

2 """
3 Created on Tue Oct 22 14:16:19 2019
4
5 @author: ahmad agung tawakkal
6 """
7
8 def NPM9(npm) :
9
10     i=0
11     NPM = input("NPM : ")
12     while i < 1:
13         if len(NPM) < 7:
14             print("NPM kurang dari 7!")
15             NPM = input("NPM : ")
16         elif len(NPM) > 7:
17             print("NPM lebih dari 7!")
18             NPM = input("NPM : ")
19         else :
20             i=1
21
22     A=NPM[0]
23     B=NPM[1]
24     C=NPM[2]
25     D=NPM[3]
26     E=NPM[4]
27     F=NPM[5]
28     G=NPM[6]
29
30     X=1
31
32     for this in A,B,C,D,E,F,G:
33
34         if int(this)%2==1:
35             print(this,end=" ")
36
37 NPM9(npm)

```

Soal 10

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Tue Oct 22 14:16:19 2019
4
5 @author: ahmad agung tawakkal
6 """
7
8 def NPM10(npm) :
9
10     i=0
11     NPM = input("NPM : ")
12     while i < 1:
13         if len(NPM) < 7:
14             print("NPM kurang dari 7!")
15             NPM = input("NPM : ")
16         elif len(NPM) > 7:
17             print("NPM lebih dari 7!")
18             NPM = input("NPM : ")

```

```

19         else :
20             i=1
21
22         A=NPM[0]
23         B=NPM[1]
24         C=NPM[2]
25         D=NPM[3]
26         E=NPM[4]
27         F=NPM[5]
28         G=NPM[6]
29
30         for X in A,B,C,D,E,F,G:
31             if int(X) > 1:
32                 for i in range(2,int(X)):
33                     if (int(X) % i) == 0:
34                         break
35             else :
36                 print(int(X),end =")")
37
38 NPM10(npm)

```

Soal 11

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Oct 22 14:16:19 2019
4
5  @author: ahmad agung tawakkal
6  """
7  #fungsi 1
8  def NPM1(npm):
9      print(" *** **          ****      ***  ***  *****      ***  *****")
10     print(" *** **      ***      ***      ***  ***  ***  ***  ***  *****")
11     print(" *** **      ***      ***      ***  ***  ***  ***  ***  ***  ***")
12     print(" *** **          ****          *****      ***  ***  ***  *****")
13     print(" *** **          ****          *****      ***  ***  ***  *****")
14     print(" *** **      ***      ***      ***          ***  ***  ***  ***      ***")
15     print(" *** **      ***      ***      ***          ***  ***  ***  ***  *****")
16     print(" *** **          ****          ***          *****      ***  *****")
17
18  #fungsi 2
19  def NPM2(npm):
20      npm = int (npm)
21      Tld=abs(npm)%100
22      for i in range(Tld):
23          print ("Hallo , ",npm," apa kabar ?")
24
25  #fungsi 3
26  def NPM3(npm):

```

```

27     for i in range(int (str(npm)[4]) + int (str(npm)[5]) + int (str
28         (npm)[6])):
29         print ("Halo, " + str(npm)[4] + str(npm)[5] + str(npm)[6] +
30             " apa kabar ?")
31
32 i = 0
33 npm = input ("Masukan NPM : ")
34 while i < 1:
35     if len (npm) < 7:
36         print ("NPM Kurang dari 7 digit")
37         npm = input ("Masukan NPM : ")
38     elif len (npm) > 7:
39         print ("NPM lebih dari 7 digit")
40         npm = input ("Masukan NPM : ")
41     else :
42         i = 1
43
44 #fungsi 4
45 def NPM4(npm):
46     NPM = input ("Masukan NPM : ")
47     print ("Halo, ",NPM[4], " apa kabar ?")
48
49 #fungsi 5
50 def NPM5(npm):
51     i=0
52     NPM = input("NPM : ")
53     while i<1:
54         if len(NPM)<7:
55             print("NPM kurang dari 7!")
56             NPM = input("Npm : ")
57         elif len(NPM)>7:
58             print("NPM lebih dari 7!")
59             NPM = input("NPM11: ")
60         else :
61             i=1
62
63     A=NPM[0]
64     B=NPM[1]
65     C=NPM[2]
66     D=NPM[3]
67     E=NPM[4]
68     F=NPM[5]
69     G=NPM[6]
70
71     for this in A,B,C,D,E,F,G:
72         print(this,end = " ")
73
74 #fungsi 6
75 def NPM6(npm):
76     i=0
77     NPM = input("NPM : ")
78     while i<1:
79         if len(NPM)<7:
80             print("NPM kurang dari 7!")
81             NPM = input("NPM : ")

```

```

82         elif len(NPM)>7:
83             print("NPM lebih dari 7!")
84             NPM = input("NPM : ")
85         else:
86             i=1
87
88         A=NPM[0]
89         B=NPM[1]
90         C=NPM[2]
91         D=NPM[3]
92         E=NPM[4]
93         F=NPM[5]
94         G=NPM[6]
95
96         X=0
97
98         for this in A,B,C,D,E,F,G:
99             X+=int( this )
100         print(X)
101
102 #fungsi 7
103 def NPM7(npm):
104
105     i=0
106     NPM = input("NPM : ")
107     while i < 1:
108         if len(NPM) < 7:
109             print("NPM kurang dari 7!")
110             NPM = input("NPM : ")
111         elif len(NPM) > 7:
112             print("NPM lebih dari 7!")
113             NPM = input("NPM : ")
114         else:
115             i=1
116
117         A=NPM[0]
118         B=NPM[1]
119         C=NPM[2]
120         D=NPM[3]
121         E=NPM[4]
122         F=NPM[5]
123         G=NPM[6]
124
125         X=1
126
127         for this in A,B,C,D,E,F,G:
128             X*=int( this )
129         print(X)
130
131 #fungsi 8
132 def NPM8(npm):
133
134     i=0
135     NPM = input("NPM : ")
136     while i < 1:
137         if len(NPM) < 7:
138             print("NPM kurang dari 7!")

```

```

139     NPM = input("NPM : ")
140     elif len(NPM)>7:
141         print("NPM lebih dari 7!")
142         NPM = input("NPM : ")
143     else :
144         i=1
145
146     A=NPM[0]
147     B=NPM[1]
148     C=NPM[2]
149     D=NPM[3]
150     E=NPM[4]
151     F=NPM[5]
152     G=NPM[6]
153
154     X=1
155
156     for this in A,B,C,D,E,F,G:
157
158         if int(this)%2==0:
159             if int(this)==0:
160                 this=""
161                 print(this,end=" ")
162
163 #fungsi 9
164 def NPM9(npm) :
165
166     i=0
167     NPM = input("NPM : ")
168     while i < 1:
169         if len(NPM) < 7:
170             print("NPM kurang dari 7!")
171             NPM = input("NPM : ")
172         elif len(NPM) > 7:
173             print("NPM lebih dari 7!")
174             NPM = input("NPM : ")
175         else :
176             i=1
177
178         A=NPM[0]
179         B=NPM[1]
180         C=NPM[2]
181         D=NPM[3]
182         E=NPM[4]
183         F=NPM[5]
184         G=NPM[6]
185
186         X=1
187
188         for this in A,B,C,D,E,F,G:
189
190             if int(this)%2==1:
191                 print(this,end=" ")
192
193 #fungsi 10
194 def NPM10(npm) :
195

```

```

196     i=0
197     NPM = input("NPM : ")
198     while i < 1:
199         if len(NPM) < 7:
200             print("NPM kurang dari 7!")
201             NPM = input("NPM : ")
202         elif len(NPM) > 7:
203             print("NPM lebih dari 7!")
204             NPM = input("NPM : ")
205         else :
206             i=1
207
208     A=NPM[0]
209     B=NPM[1]
210     C=NPM[2]
211     D=NPM[3]
212     E=NPM[4]
213     F=NPM[5]
214     G=NPM[6]
215
216     for X in A,B,C,D,E,F,G:
217         if int(X) > 1:
218             for i in range(2,int(X)):
219                 if (int(X) % i) == 0:
220                     break
221             else :
222                 print(int(X),end =" ")

```

Soal 12

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Oct 22 14:16:19 2019
4
5  @author: ahmad agung tawakkal
6  """
7
8  import lib3
9
10 class NPM:
11     def __init__(self, npm):
12         self.npm = npm
13     def NPM1(self):
14         return lib3.NPM1(self.npm)
15     def NPM2(self):
16         return lib3.NPM2(self.npm)
17     def NPM3(self):
18         return lib3.NPM3(self.npm)
19     def NPM4(self):
20         return lib3.NPM4(self.npm)
21     def NPM5(self):
22         return lib3.NPM5(self.npm)
23     def NPM6(self):
24         return lib3.NPM6(self.npm)
25     def NPM7(self):
26         return lib3.NPM7(self.npm)
27     def NPM8(self):

```



```

28         return lib3.NPM8(self.npm)
29     def NPM9(self):
30         return lib3.NPM9(self.npm)
31     def NPM10(self):
32         return lib3.NPM10(self.npm)

1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Oct 22 14:16:19 2019
4
5  @author: ahmad agung tawakkal
6  """
7
8  import kelas3lib
9
10 npm = input("Masukan NPM : ")
11 kelas = kelas3lib.NPM(npm)
12 fungsi = kelas.NPM1() #NPM1 adalah fungsi jadi dapat memanggil
    fungsi NPM yang lain
13
14 print("")

```