



# Pemrograman Berbasis Fungsi - RA

## TA Genap 2021/2022

Lecturer: Riksa Meidy Karim , S.Kom. , M.Si., M.Sc.

NAMA : MUHAMMAD DHONI APRIYADI

---

NIM : 120450111

---

---

## Tugas Exercise

---

**>> Exercise 1 >>**

```
Get Started PBF_Exercise_1_MUHAMMAD DHONI APRIYADI_Simple Pass Encryption.py > decrypt

Users > muhammaddhoniapriyadi > TUGAS PBF SEBELUM UTS > Exercise 1 > PBF_Exercise_1_MUHAMMAD DHONI APRIYADI_Simple Pass Encryption.py > decrypt

1 #Author : MUHAMMAD DHONI APRIYADI
2 #NIM : 128450111
3 #Affiliation : Sains Data ITERA
4 #Program Description : Simple Pass Encrypt
5
6 def encrypt(pw):
7     splitpass = list(pw) #mengubah input password menjadi bentuk list
8     asciipass = list()
9     for char in splitpass:
10         ascicchar = ord(char) #mengubah setiap karakter password menjadi nomor ASCII
11         ascipass.append(ascicchar) #menambahkan hasil pengubahan kedalam variabel 'ascipass'
12
13     encryptedpass = ""
14     #mengubah setiap nomor ASCII kemudian membaginya menjadi tiga buah value yg berbeda
15     for num in ascipass:
16         val_1 = num//26 + 80 #nilai pertama menghasilkan 'faktor pembagi' yang telah ditambahkan 80
17         val_2 = num%26 + 80 #nilai kedua menghasilkan modulus yang telah ditambahkan 80
18         if val_1 > val_2: #nilai ketiga
19             val_3 = '+'
20         else:
21             val_3 = '-'
22
23     encryptedpass = encryptedpass + chr(val_1) + chr(val_2) + val_3
24     #chr(string) akan mengubah nomor menjadi karakter sesuai ketentuan ASCII
25     #hasilnya tinggal digabung dan ditambahkan ke variabel 'encryptedpass'
26
27
28 return encryptedpass #akan mengembalikan password yang telah dienkripsi
29
30
31

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Simple Pass Encryption.py"
mengenkripsi password
R]~TV-Sc-Ts+Qc-Sc-TV-Sc-Ts+Se-Sg-TZ-Sf-Sc-T[-Qh-Qf-Qh-Qf-
mendekripsi password
Anak-anakcerdas2020
(base) muhammaddhoniapriyadi@192 Exercise 1 % cd "/Users/muhammaddhoniapriyadi/TUGAS PBF SEBELUM UTS/Exercise 1"; /usr/bin/env /usr/local/bin/python3 /Users/muhammaddhoniapriyadi/.vscode/extensions/ms-python.python-2021.12.1559732655/pythonFiles/lib/python/debugpy/launcher 57093 -- "/Users/muhammaddhoniapriyadi/TUGAS PBF SEBELUM UTS/Exercise 1/PBF_Exercise_1_MUHAMMAD DHONI APRIYADI_Simple Pass Encryption.py"
mengenkripsi password
R]~TV-Sc-Ts+Qc-Sc-TV-Sc-Ts+Se-Sg-TZ-Sf-Sc-T[-Qh-Qf-Qh-Qf-
mendekripsi password
Anak-anakcerdas2020
(base) muhammaddhoniapriyadi@192 Exercise 1 % []
Python 3.10.1 |Anaconda, Inc.| (base) ~
```

```
Get Started PBF_Exercise_1_MUHAMMAD DHONI APRIYADI_Simple Pass Encryption.py > decrypt

Users > muhammaddhoniapriyadi > TUGAS PBF SEBELUM UTS > Exercise 1 > PBF_Exercise_1_MUHAMMAD DHONI APRIYADI_Simple Pass Encryption.py > decrypt

53     for i in ascipass: #tinggal mengubah setiap nomor ASCII di ascipass menjadi karakter dasar +
54         char = chr(i)
55         password = password + char
56
57     return password #akan mengembalikan password yang telah di-decrypt
58
59
60
61
62 print('mengenkripsi password')
63 x = 'Anak-anakcerdas2020'
64 hasilencrypt = encrypt(x)
65 print(hasilencrypt)
66
67 print('_'* 30)
68 print('mendekripsi password')
69 y = [R]~TV-Sc-Ts+Qc-Sc-TV-Sc-Ts+Se-Sg-TZ-Sf-Sc-T[-Qh-Qf-Qh-Qf-
70 print(decrypt(y))

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Simple Pass Encryption.py"
mengenkripsi password
R]~TV-Sc-Ts+Qc-Sc-TV-Sc-Ts+Se-Sg-TZ-Sf-Sc-T[-Qh-Qf-Qh-Qf-
mendekripsi password
Anak-anakcerdas2020
(base) muhammaddhoniapriyadi@192 Exercise 1 % cd "/Users/muhammaddhoniapriyadi/TUGAS PBF SEBELUM UTS/Exercise 1"; /usr/bin/env /usr/local/bin/python3 /Users/muhammaddhoniapriyadi/.vscode/extensions/ms-python.python-2021.12.1559732655/pythonFiles/lib/python/debugpy/launcher 57093 -- "/Users/muhammaddhoniapriyadi/TUGAS PBF SEBELUM UTS/Exercise 1/PBF_Exercise_1_MUHAMMAD DHONI APRIYADI_Simple Pass Encryption.py"
mengenkripsi password
R]~TV-Sc-Ts+Qc-Sc-TV-Sc-Ts+Se-Sg-TZ-Sf-Sc-T[-Qh-Qf-Qh-Qf-
mendekripsi password
Anak-anakcerdas2020
(base) muhammaddhoniapriyadi@192 Exercise 1 % []
Python 3.10.1 |Anaconda, Inc.| (base) ~
```

## >> Exercise 2 >>

The screenshot shows a development environment on repl.it. On the left, the 'Files' sidebar lists 'main.py', 'bildua.txt', and 'bilsatu.txt'. Below it, 'Packager files' include 'poetry.lock' and 'pyproject.toml'. The main workspace contains the following code:

```
1 #Author : MUHAMMAD DHONI APRIYADI
2 #NIM : 120450111
3 #Affiliation : Sains Data ITERA
4 #Program Description : Penjumlahan dua bilangan
5
6
7 firstfile = open('bilsatu.txt', 'r')
8 secfile = open('bildua.txt')
9
10 firstvalstr = firstfile.read()
11 secvalstr = secfile.read()
12
13 firstval = int(firstvalstr)
14 secval = int(secvalstr)
15
16 addition = (lambda x,y: x+y)
17 print(addition(firstval, secval))
18
```

The 'Console' tab on the right shows the output of the program: `1161863033755481508524753707463953874904342178407580049143859624033  
263763736576488440596584806646855`.

## >> Exercise 3 >>

```

# Author : MUHAMMAD DHONI APIYADI NIM : 120450111 Affiliation : Sains Data ITERA Program Description : NON Linear Pendulum

#Author : MUHAMMAD DHONI APIYADI
#NIM : 120450111
#Affiliation : Sains Data ITERA
#Program Description : Simple Football Game
def euler_(t,h,y,dy,func):
    d2y = func(t,y,dy)
    y_next = y + (h * dy)
    dy_next = dy + (h * d2y)
    return (y_next, dy_next)

[2] def euler_cromer_(t,h,y,dy,func):
    d2y = func(t, y, dy)
    dy_next = dy + (h * d2y)
    y_next = y + (h * dy_next)

    return (y_next, dy_next)

[3] # Euler Method
def cauchy_euler(params,Func):
    # Initial Condition
    t0 = params['t0']
    t_akhir = params['t_akhir']
    h = params['h']
    y0 = params['y0']
    dy0 = params['dy0']

    res_euler_ = []
    t = []
    step = int((t_akhir - t0) / h)

    for i in range(step):
        res_euler_.append(y0)
        t.append(t0)
        y0 = y0 + dy0
        dy0 = dy0 + h * Func(t0, y0, dy0)
        t0 = t0 + h

    return (t,res_euler_)

def cauchy_eulercromer(params,Func):
    # Initial Condition
    t0 = params['t0']
    t_akhir = params['t_akhir']
    h = params['h']
    y0 = params['y0']
    dy0 = params['dy0']

    res_euler_cromer_ = []
    t = []
    step = int((t_akhir - t0) / h)

    for i in range(step):
        res_euler_cromer_.append(y0)
        t.append(t0)
        y0 = y0 + dy0
        dy0 = dy0 + h * Func(t0, y0, dy0)
        t0 = t0 + h

```

File Edit View Insert Runtime Tools Help All changes saved

```
+ Code + Text
# MAIN i Program
from math import *
import matplotlib.pyplot as plt

g = 9.81
l = Loading...
k = 0
u0 = 0.5 * pi
du0 = 0
t0 = 0
t_akhir = 4
h = 0.01
w0 = g/l

def function(t,u,du):
    return w0 * sin(u) - k*u

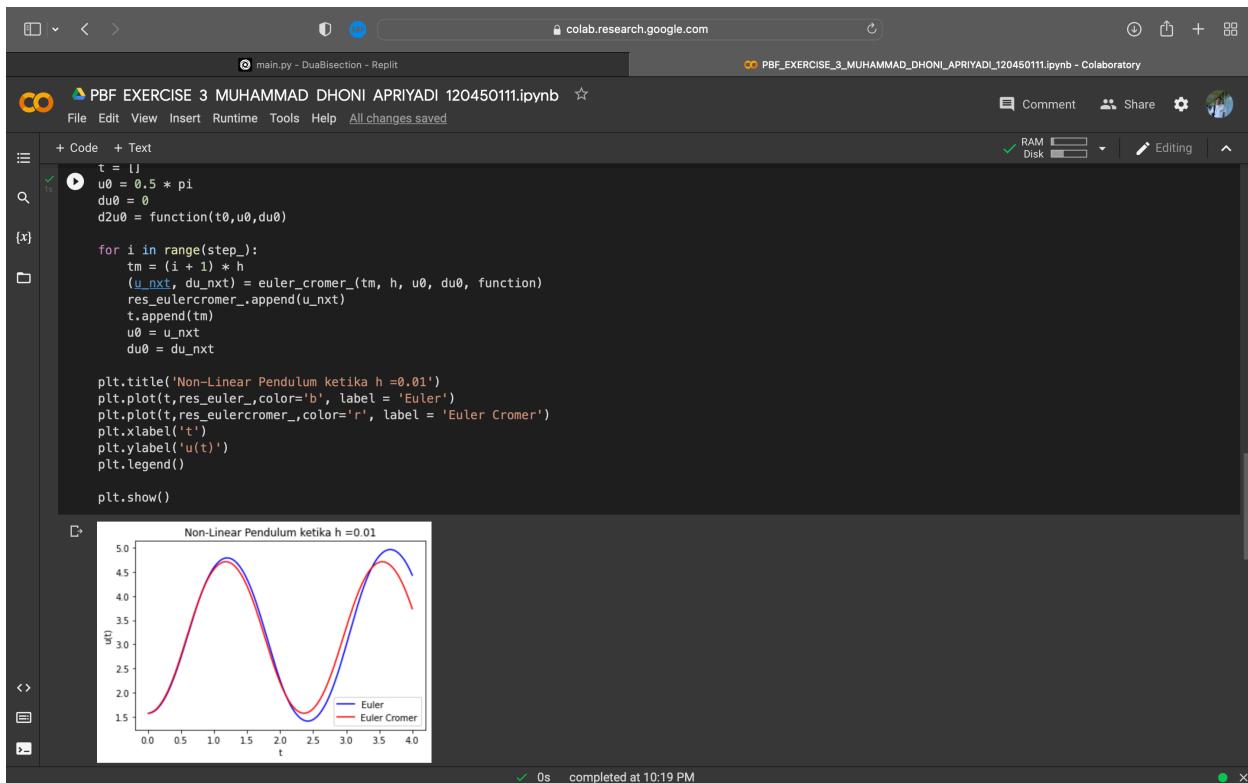
res_euler_ = []
res_eulercromer_ = []
t = []
step_ = int((t_akhir - t0) / h)

for i in range(step_):
    tm = (i + 1) * h
    (u_nxt, du_nxt) = euler_(tm, h, u0, du0, function)
    res_euler_.append(u_nxt)
    t.append(tm)
    u0 = u_nxt
    du0 = du_nxt

t = []
u0 = 0.5 * pi
du0 = 0
d2u0 = function(t0,u0,du0)

for i in range(step_):
    tm = (i + 1) * h
    (u_nxt, du_nxt) = euler_cromer_(tm, h, u0, du0, function)
    res_eulercromer_.append(u_nxt)
    t.append(tm)
    u0 = u_nxt
    du0 = du_nxt
```

0s completed at 10:19 PM



---

**>> Exercise 4 >>**

```

#Author : MUHAMMAD DHONI APIYADI
#NIM : 120450111
#Affiliation : Sains Data ITERA
#Program Description : Simple Football Game

import random
import math
#Lambda value for higher rated team dengan menggunakan distribusi poisson
lamb_One = 1.148698355
#Lambda value for lower rated team
lamb_Two = 0.8705505633

#Distribusi Poisson digunakan dalam menghitung gol yang dicetak oleh Home (tuanrumah)
def homeMatch(home_Rating,away_Rating):
    global lamb_One
    global x
    global y
    if x == y:
        raise ValueError
    else:
        lamb = lamb_One**((int(home_Rating)-int(away_Rating)))
        homeScore = 0
        z = random.random()
        while z > 0:
            z = z - ((lamb*homeScore * math.exp(lamb * -1))/(math.factorial(homeScore)))
            homeScore += 1
        return (homeScore-1)

#Distribusi Poisson menghitung gol yang dicetak oleh tim tandang (away)
def awayMatch(home_Rating,away_Rating):
    global lamb_Two
    global x
    global y

```

```

awayScore = 0

#Permainan dimulai dengan menggunakan range
for x in range(league_Size):
    print("====")
    print(team_Names[x]+"'s home games: ")
    print("====\n")
    for y in range(league_Size):
        error = 0
        try:
            homeScore = homeMatch(team_Skill[x],team_Skill[y])
            #Skipping a game to stop a team playing itself
            except ValueError:
                pass
            error += 1
        try:
            awayScore = awayMatch(team_Skill[x],team_Skill[y])
            except ValueError:
                pass
        if error == 0:
            #memperbarui dan updating lists
            print(team_Names[x],homeScore,"-",awayScore,team_Names[y],"\\n")
            team_For[x] += homeScore
            team_For[y] += awayScore
            team_Against[x] += awayScore
            team_Against[y] += homeScore
            if homeScore > awayScore:
                team_Wins[x] += 1
                team_Losses[y] += 1
                team_Points[x] += 3
            elif homeScore == awayScore:
                team_Draws[x] += 1
                team_Draws[y] += 1
                team_Points[x] += 1
                team_Points[y] += 1
            else:
                team_Wins[y] += 1

```

The screenshot shows a Google Colab interface with a dark theme. The top bar includes the URL 'colab.research.google.com' and the title 'PBF Exercise4 SimpleFBG MUHAMMAD DHONI\_APRIYADI 120450111.ipynb'. The code cell contains Python code for a football league simulation:

```
#Printing table (unsorted)
print("Hasil Akhir : ")
for x in range(league_Size):
    #Lots of formatting
    print(team_Names[x]+(15-len(team_Names[x]))*" "+" Skill: "+str(team_Skill[x])+(5-len(str(team_Skill[x])))*)" "+" Points: "+str(team_Points[x])+(5-len(str(team_Points[x])))*" "+" Wins: "+str(team_Wins[x])+(5-len(str(team_Wins[x])))*" "+" Draws: "+str(team_Draws[x])+(5-len(str(team_Draws[x])))*" "+" Losses: "+str(team_Losses[x])+(5-len(str(team_Losses[x])))*" "
team_Points.sort()
print(team_Points)
```

The output window shows the user input and the generated league table:

```
Masukkan jumlah tim yang bertanding di Liga: 2
Enter team 1 name: ABC
Enter team 2 name: EFG
Enter ABC skill: 5
Enter EFG skill: 6
=====
ABC's home games:
=====
ABC 0 - 1 EFG
=====
EFG's home games:
=====
EFG 1 - 2 ABC
Hasil Akhir :
ABC      Skill: 5      Points: 3      For: 2      Against: 2      Goal difference: 0      Wins: 1      Draws: 0      Losses: 1
EFG      Skill: 6      Points: 3      For: 2      Against: 2      Goal difference: 0      Wins: 1      Draws: 0      Losses: 1
[3, 3]
```