

01153 - ALGORITMA DAN STRUKTUR DATA 1

SUB PROGRAM

OLEH : ANDRI HERYANDI, M.T.



08

SUB PROGRAM

01153 - Algoritma dan Struktur Data 1

■ Part #1

- Definisi sub program
- Algoritma
 - Procedure
 - Pendeklarasian Procedure
 - Pemanggilan Procedure
 - Function
 - Pendeklarasian Function
 - Pemanggilan Function
- Python
 - Deklarasi Function
 - Pemanggilan Function
 - Urutan Penulisan Argument
 - Built-in Function

■ Part #2

- Default Value
- Passing Argument
- Recursive Function
- Module
 - Membuat Module
 - Import Module
 - `import` Modules
 - `import` Modules `as`
 - `from` Modules `import`
 - Menggunakan Module
 - Install Module
 - UnInstall Module



Oleh : Andri Heryandi, M.T.

01153 - ALGORITMA DAN STRUKTUR DATA 1

SUB PROGRAM (PART #2)

OLEH : ANDRI HERYANDI, M.T.



RECURSIVE FUNCTION

01153 - Algoritma dan Struktur Data 1

- Recursive function adalah function yang didalamnya memanggil function itu sendiri.
- Contoh klasik recursive function banyak ditemukan di ilmu matematika.
- Contoh kasus recursive function adalah factorial, dimana factorial didefinisikan sebagai berikut :

$$N! = N * (N - 1)!$$

dimana $0! = 1$

- Contoh :

```
4! = 4 * 3!
      3 * 2!
          2 * 1!
              1 * 0!
                  1 * 1 => return 1
              2 * 1 => return 2
          3 * 2 => return 6
      4 * 6 => return 24
```



RECURSIVE FUNCTION

01153 - Algoritma dan Struktur Data 1

- Ada hal-hal yang harus diperhatikan ketika membuat function recursive yaitu :
 1. Adanya pemanggilan ke function itu sendiri
 2. Adanya base case (kondisi yang akan mengakhiri proses rekursif). Dalam factorial, base case-nya adalah $0! = 1$. Jika anda tidak menyertakan base case, maka kemungkinan besar akan menghasilkan eksekusi yang tidak diharapkan.



RECURSIVE FUNCTION

01153 - Algoritma dan Struktur Data 1

■ Contoh program python dengan recursive function (Factorial)

```
def faktorial(N):  
    if N<=1:  
        return 1  
    else:  
        return N * faktorial(N-1)  
  
print("0! = ", faktorial(0))  
print("1! = ", faktorial(1))  
print("2! = ", faktorial(2))  
print("3! = ", faktorial(3))  
print("4! = ", faktorial(4))  
print("5! = ", faktorial(5))
```

Hasil eksekusi

0!	=	1
1!	=	1
2!	=	2
3!	=	6
4!	=	24
5!	=	120



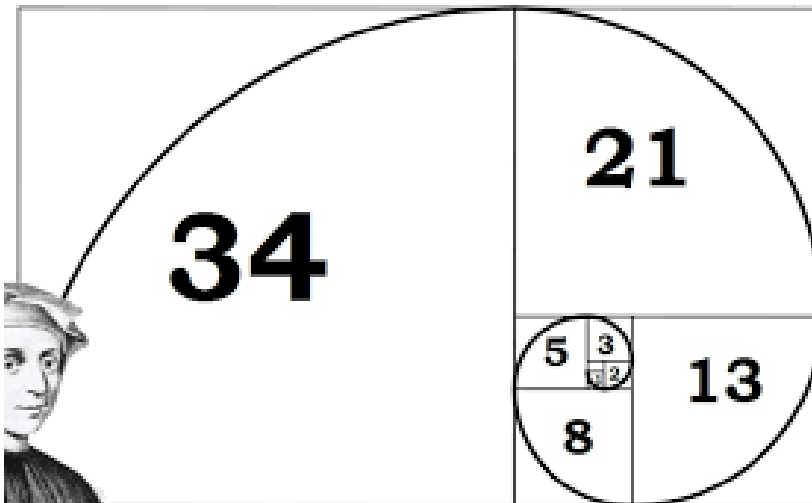
RECURSIVE FUNCTION

01153 - Algoritma dan Struktur Data 1

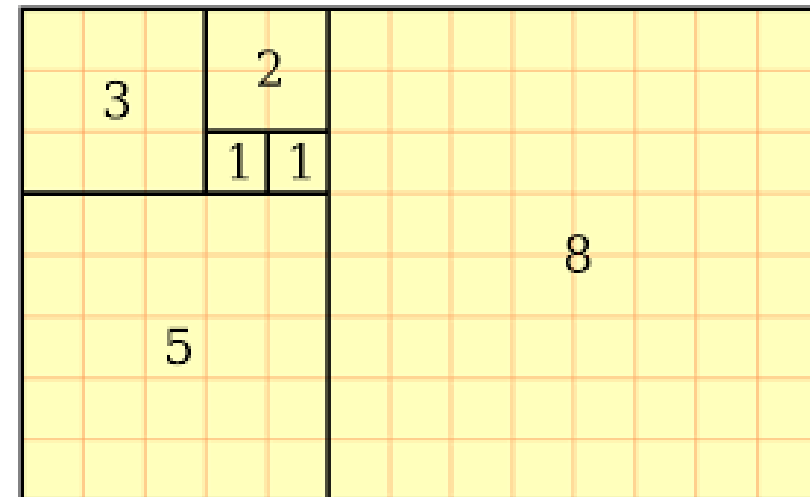
- Contoh lain kasus recursive function
- Deret Fibonacci adalah deret bilangan yang dihasilkan dengan menjumlahkan 2 suku sebelumnya, dimana $Fibonacci(0) = 0$ dan $Fibonacci(1) = 1$

$$Fibonacci(N) = Fibonacci(N - 2) + Fibonacci(N - 1)$$

- Contoh deret Fibonacci adalah : 0 1 1 2 3 5 8 13 21 34 dst



Oleh : An



RECURSIVE FUNCTION

01153 - Algoritma dan Struktur Data 1

■ Contoh program python dengan recursive function (Fibonacci)

```
def fibonacci(N):  
    if N<=1:  
        return N  
    else:  
        return fibonacci(N-2)+fibonacci(N-1)  
  
for i in range(0,11):  
    print(fibonacci(i), end=" ")
```

Hasil eksekusi

0 1 1 2 3 5 8 13 21 34 55



MODULE

01153 - Algoritma dan Struktur Data 1

- Definisi dari module dalam python adalah
 - Module adalah sebuah file yang definisi dan pernyataan
 - Module adalah code library (pustaka kode)
- Module boleh berisi kumpulan variable, function (/procedure), class dan lain-lain
- Nama suatu module diambil dari nama filenya
- Untuk menggunakan suatu module, maka module tersebut harus diimport terlebih dahulu.
- Ketika sebuah function atau objek dari suatu module akan digunakan, maka harus diawali dengan nama module-nya.



MEMBUAT MODULE

01153 - Algoritma dan Struktur Data 1

- Buat sebuah file python yang diberi nama sesuai dengan nama module yang diinginkan.
Contoh : jika anda ingin module bernama **my_module** maka buatlah file **my_module.py**

Nama File : my_module.py

```
# pendefinisian objek
pi = 22/7

# pendefinisian function
def luasLingkaran(radius):
    return pi * radius ** 2

def kelilingLingkaran(radius):
    return 2 * pi * radius

def kabisat(tahun):
    return (tahun % 100 == 0) and (tahun % 400 == 0) \
        or \
        (tahun % 100 != 0) and (tahun % 4 == 0)
```



IMPORT MODULE

01153 - Algoritma dan Struktur Data 1

- Sebelum definisi atau function yang berada dalam module digunakan, maka module tersebut harus diimport dulu dengan menggunakan keyword **import**.

Nama File : test_module.py

```
import my_module

print("PI = ", my_module.pi)
print("Luas Lingkaran : ", my_module.luasLingkaran(10))
print("Keliling Lingkaran : ", my_module.kelilingLingkaran(10))
print("Tahun 1995 adalah tahun kabisat ? ", my_module.kabisat(1995))
print("Tahun 2000 adalah tahun kabisat ? ", my_module.kabisat(2000))
```

Hasil eksekusi

```
PI = 3.142857142857143
Luas Lingkaran : 314.2857142857143
Keliling Lingkaran : 62.857142857142854
Tahun 1995 adalah tahun kabisat ? False
Tahun 2000 adalah tahun kabisat ? True
```



IMPORT MODULE

01153 - Algoritma dan Struktur Data 1

- Sebuah module yang diimport boleh direname (ganti nama). Gunakan keyword **as** untuk mendefinisikan alias (nama lain) dari module yang anda import.
- Dengan merename module, maka anda bisa memanggil module dengan lebih sederhana dan lebih singkat.

Nama File : test_module.py

```
import my_module as my

print("PI = ", my.pi)
print("Luas Lingkaran : ", my.luasLingkaran(10))
print("Tahun 2000 adalah tahun kabisat ? ", my.kabisat(2000))
```

Hasil eksekusi

```
PI = 3.142857142857143
Luas Lingkaran : 314.2857142857143
Tahun 2000 adalah tahun kabisat ? True
```



IMPORT MODULE

01153 - Algoritma dan Struktur Data 1

- Ada juga cara variasi cara import module menggunakan keyword **from import**
- Dengan keyword ini anda bisa mengimport function-function tertentu saja.
- Dengan import seperti ini, function cukup dipanggil dengan namanya (tanpa menyebutkan nama modulnya)

Nama File : test_module.py

```
from my_module import luasLingkaran, kabisat
# gunakan "from my_module import * " jika ingin import semua objek/function [kurang disarankan]
print("Luas Lingkaran : ", luasLingkaran(10))
print("Tahun 2000 adalah tahun kabisat ? ", kabisat(2000))
```

Hasil eksekusi

```
PI = 3.142857142857143
Tahun 2000 adalah tahun kabisat ? True
```



Oleh : Andri Heryandi, M.T.

Catatan:

Anda tidak disarankan menggunakan **"from my_module import * "** karena akan membingungkan anda ketika sebuah function yang bernama sama berada pada module yang berbeda

BUILT-IN MODULE

01153 - Algoritma dan Struktur Data 1

- Built-in module adalah module-module yang sudah terinstall ketika anda menginstall python.
- Daftar built-in module di python dapat diakses di <https://docs.python.org/3.8/py-modindex.html>
- Untuk mengetahui module yang ada, langkahnya adalah :
 - Buka IDLE
 - `>>> help()` → Masuk ke mode help. Nanti akan memunculkan help prompt (`help>>>`)
 - `help>>> modules` → Menampilkan module-module yang bisa anda pakai
 - `help>>> math` → Melihat isi module math
 - `help>>> modules math` → Mencari module yang mengandung kata math
 - `help>>> quit` → Keluar dari mode help



INSTALL MODULE

01153 - Algoritma dan Struktur Data 1

- Jika anda membutuhkan module lain yang bukan merupakan built-in module, maka anda harus menginstallnya terlebih dahulu.
- Untuk menginstall module baru, maka anda akan menggunakan sebuah program bawaan yaitu Python package manager yang dikenal dengan nama pip
- Module-module biasanya tersusun dalam bentuk paket (package).
- Sintak untuk melakukan install adalah :

```
pip install <nama_package>
```

- Ada beberapa package yang “wajib” diinstall untuk seorang data scientist diantaranya numpy, pandas, dan matplotlib.



INSTALL MODULE

01153 - Algoritma dan Struktur Data 1

- Numpy (Numerical Python) adalah library Python yang fokus pada komputasi saintifik. *NumPy* memiliki kemampuan untuk membentuk dan mengelola objek N-dimensional array.
- Langkah menginstall package (numpy) adalah :
 - Masuk ke command prompt
 - Di command prompt tulis : “pip install numpy”
 - Tunggu sampai proses install selesai.

```
C:\Windows\system32>pip install numpy
Collecting numpy
  Downloading numpy-1.21.5-cp38-cp38-win32.whl (11.7 MB)
    |████████████████████████████████████████| 11.7 MB 1.7 MB/s
Installing collected packages: numpy
Successfully installed numpy-1.21.5
```

```
C:\Windows\system32>
```



INSTALL MODULE

01153 - Algoritma dan Struktur Data 1

- Pandas adalah Pustaka (library) perangkat lunak yang ditulis untuk bahasa pemrograman Python untuk manipulasi dan analisis data.
- Langkah menginstall package pandas adalah :
 - Masuk ke command prompt
 - Di command prompt tulis : “pip install pandas”
 - Tunggu sampai proses install selesai.

```
C:\Windows\system32>pip install pandas
Collecting pandas
  Downloading pandas-1.3.5-cp38-cp38-win32.whl (9.1 MB)
    |████████████████████████████████████████| 9.1 MB 1.6 MB/s
Requirement already satisfied: numpy>=1.17.3 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from pandas) (1.21.5)
Collecting pytz>=2017.3
  Downloading pytz-2021.3-py2.py3-none-any.whl (503 kB)
    |████████████████████████████████████████| 503 kB 2.2 MB/s
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from python-dateutil>=2.7.3->pandas) (1.16.0)
Installing collected packages: pytz, pandas
Successfully installed pandas-1.3.5 pytz-2021.3
C:\Windows\system32>
```



Oleh : Andri Heryandi, M.T.

INSTALL MODULE

01153 - Algoritma dan Struktur Data 1

- Matplotlib adalah pustaka (library) untuk menggambar (plot) untuk bahasa pemrograman Python dan ekstensi matematika numeriknya NumPy.
- Langkah menginstall package pandas adalah :
 - Masuk ke command prompt
 - Di command prompt tulis : “pip install matplotlib”
 - Tunggu sampai proses install selesai.

```
C:\Windows\system32>pip install matplotlib
Collecting matplotlib
  Downloading matplotlib-3.5.1-cp38-cp38-win32.whl (7.1 MB)
    |████████████████████████████████████████| 7.1 MB 2.2 MB/s
Requirement already satisfied: cycler>=0.10 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (8.4.0)
Requirement already satisfied: numpy>=1.17 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (1.21.5)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: packaging>=20.0 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (21.3)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (4.28.3)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (3.0.6)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\andri heryandi\appdata\local\programs\python\python38-32\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Installing collected packages: matplotlib
Successfully installed matplotlib-3.5.1
C:\Windows\system32>
```



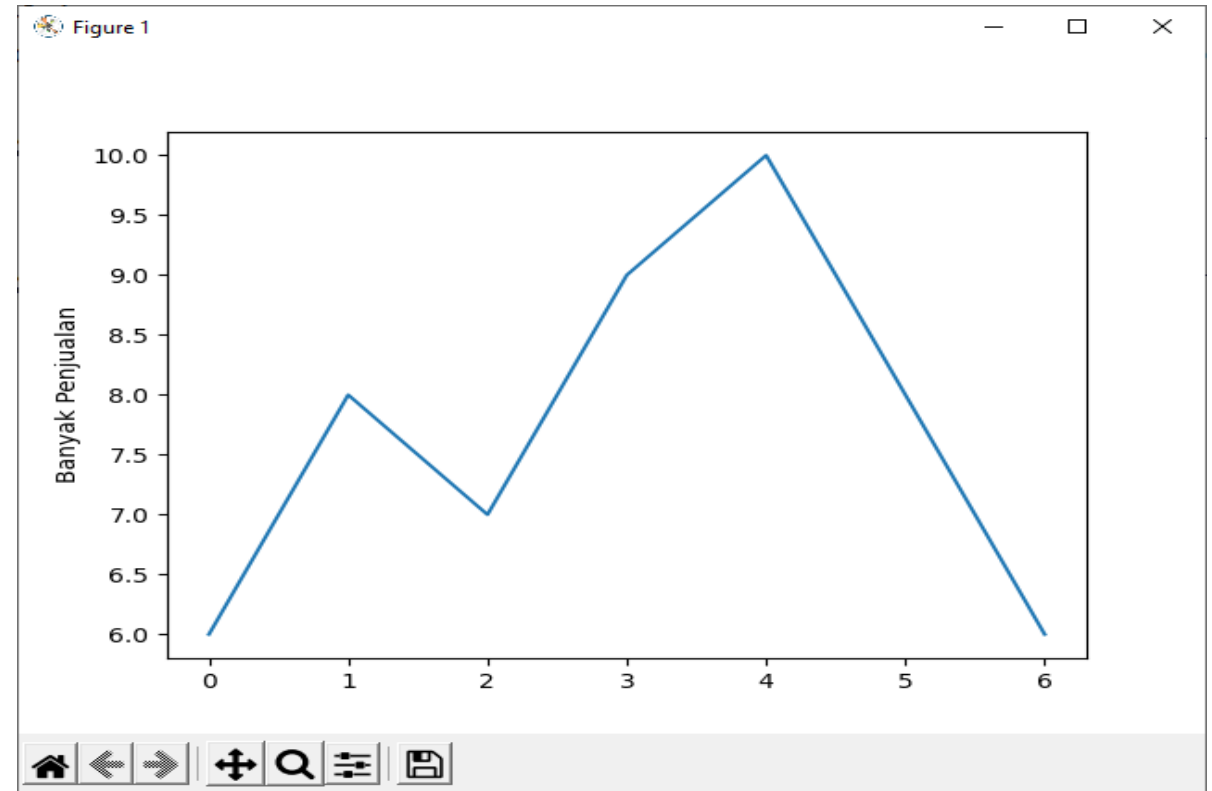
INSTALL MODULE

01153 - Algoritma dan Struktur Data 1

■ Test module yang baru diinstall

```
import matplotlib.pyplot as plt
plt.plot([6, 8, 7, 9, 10, 8, 6])
plt.ylabel("Banyak Penjualan")
plt.show()
```

Hasil eksekusi



UNINSTALL MODULE

01153 - Algoritma dan Struktur Data 1

- Jika anda tidak membutuhkan lagi suatu module/package,
- Sintak untuk melakukan uninstall adalah :

```
pip uninstall <nama_package>
```

- Jika ada pertanyaan “Proceed (Y/n)? “, jawab Y untuk memulai proses uninstall.

