

# 200411100085\_Modul1\_Review Algoritma

March 29, 2022

## 1 Konsep

### 1.1 Iterasi, syntax for dan while

Iterasi, syntax for dan while, jelaskan perbedaan kedua syntax, berikan contoh masing-masing penggunaan

Syntax for digunakan untuk melakukan iterasi dari sebuah nilai. Biasanya syntax for digunakan untuk perulangan yang pasti. contoh penggunaan dalam menampilkan bilangan genap :

```
for i in range(0,2,11) :  
    print(i)
```

Syntax while digunakan untuk melakukan perulangan juga tetapi untuk syntax while memerlukan suatu kondisi yang memenuhi agar dapat berjalan. Dan syntax while biasanya tidak memiliki nilai pasti untuk perulangannya. Contoh penggunaan syntax while :

```
lanjut = True
```

```
while lanjut :  
    jwb = input("=> Apakah anda ingin lanjut y/t : ")  
    if jwb == "t" :  
        lanjut = False
```

### 1.2 List

List, apa itu list, bagaimana mengakses setiap anggota di dalam list, berikan contoh penggunaan.

List adalah tipe data di python yang digunakan untuk menyimpan banyak nilai dan bisa menyimpan nilai dengan bertipe data berbeda di dalam satu variabel.

```
mahasiswa = ["Alexandra",90,"Steve Leo",80,"Ramon Gonzalez",76]
```

Untuk pengaksesan list menggunakan no indeks. contoh :

```
mahasiswa = ["Alexandra",90,"Steve Leo",80,"Ramon Gonzalez",76]
```

```
print("""  
Mahasiswa 1 = %s, Nilai = %d  
Mahasiswa 2 = %s, Nilai = %d  
Mahasiswa 3 = %s, Nilai = %d
```

```

""" %
(mahasiswa[0],mahasiswa[1],mahasiswa[2],mahasiswa[3],mahasiswa[4],mahasiswa[5]) )

contoh penggunaan list bisa digunakan untuk membuat daftar mahasiswa :

namaMahasiswa = ["Alexandra","Jaka","Budi"]
nimMahasiswa = ["21801111001","21801111002","21801111003"]
umurMahasiswa = [17,19,18]

print("""
-----
| Daftar Mahasiswa |
-----

Mahasiswa 1
+-----+
Nama = %s
NIM = %s
Umur = %d
+-----+

Mahasiswa 2
+-----+
Nama = %s
NIM = %s
Umur = %d
+-----+

Mahasiswa 3
+-----+
Nama = %s
NIM = %s
Umur = %d
+-----+

""" %
(
    namaMahasiswa[0],nimMahasiswa[0],umurMahasiswa[0],
    namaMahasiswa[1],nimMahasiswa[1],umurMahasiswa[1],
    namaMahasiswa[2],nimMahasiswa[2],umurMahasiswa[2],
)
)

```

### 1.3 Fungsi

Sebutkan jenis fungsi, perbedaan, dan syntax dari fungsi, berikan contoh penggunaan.

Jenis Fungsi :

### - Fungsi biasa tanpa parameter

Fungsi biasa tanpa parameter adalah potongan code yang dibuat tanpa parameter dan return value.

```
def namaFungsi() :  
    #code  
  
#contoh Menampilkan matrix 2D  
def dispMatrix() :  
    matrix = [[1,2],[3,4]]  
    for baris in matrix :  
        print(baris)
```

dispMatrix()

### - Fungsi dengan parameter

Fungsi dengan parameter adalah potongan code yang dibuat dengan parameter/argument.

```
def namaFungsi(parameter) :  
    #code  
  
#contoh Menampilkan matrix 2D  
def dispMatrix(matrix) :  
    for baris in matrix :  
        print(baris)
```

```
matrixList = [[1,2],[3,4]]  
dispMatrix(matrixList)
```

### - Fungsi dengan return value dan parameter

Fungsi dengan return value dan parameter potongan code yang dibuat dengan parameter/argument dan disertai return value.

```
def namaFungsi(parameter) :  
    #code  
    return nilai  
  
#contoh fungsi buat Matrix 1D  
def buatMatrix(size) :  
    matrixList = []  
    for i in range(size) :  
        matrixList.append(int(input("=> Masukkan Angka = ")))  
    return matrixList
```

```
matrix = buatMatrix(4)  
print(matrix)
```

### - Fungsi dengan return value ,parameter, dan nilai default

Fungsi dengan return value ,parameter, dan nilai default adalah potongan code yang dibuat dengan parameter/argument, disertai return value dan ditambah dengan nilai default.

```
def namaFungsi(parameter=nilai_default) :  
    #code  
    return nilai  
  
#contoh fungsi Buat Matrix  
def buatMatrix(size=3) :  
    matrixList = []  
    for i in range(size) :  
        matrixList.append(int(input("=> Masukkan Angka = ")))  
    return matrixList  
  
matrix = buatMatrix()  
print(matrix)
```

### - Fungsi Recursif

Fungsi Recursif adalah teknik dalam memecahkan suatu masalah menggunakan fungsi yang memanggil fungsi tersebut sendiri sampai suatu kondisi tertentu.

```
def namaFungsi(parameter) :  
    #code  
    if stoping_condition :  
        #code  
    else :  
        return namaFungsi(parameter-1)  
  
def faktorialRecursif(number):  
    if number==1 or number==0:  
        val=1  
    else:  
        val=number*faktorialRecursif(number-1)  
    return(val)  
  
num = faktorialRecursif(5)  
print(num)
```

Biasanya perbedaan penggunaan fungsi yang tidak mengembalikan nilai digunakan untuk memecahkan masalah yang ada di lingkup fungsi tersebut seperti menampilkan tulisan petunjuk, sedangkan fungsi yang mengembalikan nilai/return value akan mengembalikan suatu nilai yang dapat digunakan untuk operasi selanjutnya.

## 2 Implementasi

### 2.1 Persamaan Matematika

```
[7]: def createList(size) :  
    listNums = []  
    for i in range(size) :  
        listNums.append(int(input("=> Masukkan Angka = ")))  
    return listNums  
  
def persamaan1(listNums) :  
    total = 0  
    for num in listNums :  
        total+=num  
    result = 1/len(listNums) * total  
    return result  
  
def persamaan2(listNums,y) :  
    total = 0  
  
    for num in listNums :  
        total+=(num-y)**2  
  
    result = 1/(len(listNums)-1) * total  
    return result  
  
angka1 = createList(7)  
x = persamaan1(angka1)  
z = persamaan2(angka1,x)  
print("""  
  
Data = %s  
  
Persamaan1 = %s  
Persamaan2 = %s  
""") %  
(  
    angka1,  
    x,z  
)
```

```
=> Masukkan Angka = 10  
=> Masukkan Angka = 12  
=> Masukkan Angka = 8  
=> Masukkan Angka = 4  
=> Masukkan Angka = 9
```

```
=> Masukkan Angka = 1
=> Masukkan Angka = 7
```

```
Data = [10, 12, 8, 4, 9, 1, 7]
```

```
Persamaan1 = 7.285714285714286
Persamaan2 = 13.904761904761905
```

## 2.2 Matriks 1D

```
[8]: def buatMatrix(ukuran) :
    listMatrix = []
    for i in range(ukuran) :
        listMatrix.append(int(input("=> Masukkan Angka = ")))
    return listMatrix

def penjumlahanMatrix(matrix1,matrix2) :
    ukuranMatrix = len(matrix1) == len(matrix2)
    matrixHasil = []

    if ukuranMatrix :
        for i in range(len(matrix1)) :
            matrixHasil.append(matrix1[i]+matrix2[i])

        return matrixHasil
    else :
        return "Ukuran matrix tidak sama"

def perkalianSkalar(bil,matrix) :
    matrixHasil = []

    for mat in matrix :
        matrixHasil.append(mat*bil)

    return matrixHasil

def perkalianMatrix(matrix1,matrix2) :
    ukuranMatrix = len(matrix1) == len(matrix2)
    matrixHasil = []

    if ukuranMatrix :
        for i in range(len(matrix1)) :
            matrixHasil.append(matrix1[i]*matrix2[i])

        return matrixHasil
```

```

else :
    return "Ukuran matrix tidak sama"

matrix1 = buatMatrix(4)
matrix2 = buatMatrix(3)
matrix3 = buatMatrix(4)

jumlah1 = penjumlahanMatrix(matrix1,matrix2)
jumlah2 = penjumlahanMatrix(matrix1,matrix3)

perkalian1 = perkalianSkalar(3,jumlah2)
perkalian2 = perkalianMatrix(perkalian1,matrix1)

print("""
Matrix1 = %s
Matrix2 = %s
Matrix3 = %s

Hasil Penjumlahan1 = %s
Hasil Penjumlahan2 = %s

Hasil Perkalian Skalar = %s
Hasil Perkalian Matrix = %s

""")
(
    matrix1,matrix2,matrix3,
    jumlah1,jumlah2,
    perkalian1,perkalian2
)
)

```

```

=> Masukkan Angka = 4
=> Masukkan Angka = 1
=> Masukkan Angka = 9
=> Masukkan Angka = 10
=> Masukkan Angka = 5
=> Masukkan Angka = 2
=> Masukkan Angka = 3
=> Masukkan Angka = 6
=> Masukkan Angka = 5
=> Masukkan Angka = 1
=> Masukkan Angka = 4

```

```
Matrix1 = [4, 1, 9, 10]
Matrix2 = [5, 2, 3]
Matrix3 = [6, 5, 1, 4]
```

```
Hasil Penjumlahan1 = Ukuran matrix tidak sama
Hasil Penjumlahan2 = [10, 6, 10, 14]
```

```
Hasil Perkalian Skalar = [30, 18, 30, 42]
Hasil Perkalian Matrix = [120, 18, 270, 420]
```

## 2.3 Matriks 2D

```
[1]: # Gunakan Import module from URL jika anda tidak ingin mendownload module_
      ↳matrix.py
# Anda bisa mendownload di url :
# --- Github ---
# - https://raw.githubusercontent.com/satriaer77/struktur_data/main/Praktikum/
      ↳Praktikum1/modules/matrixLib.py (Versi OOP)
# - https://raw.githubusercontent.com/satriaer77/struktur_data/main/Praktikum/
      ↳Praktikum1/modules/matrix.py
#
# --- Google Drive ---
# - https://drive.google.com/drive/folders/1Y111X8nq6TKDqPrqs6qRL2gwnt3amx-U?
      ↳usp=sharing

#----- Import from URL -----#
# Uncomment code dibawah ini bila akan menggunakan module lewat URL
# Setelah itu beri comment pada code from 'path' import matrix

import urllib.request

urlModule = urllib.request.urlopen('https://raw.githubusercontent.com/
      ↳satriaer77/struktur_data/main/Praktikum/Praktikum1/modules/matrixLib.py')
exec(urlModule.read()) #Menjalankan file py dari urlModule
matrix      = Matrix() #Inisialisasi class Matrix

#----- End Import from URL -----#

#from modules import matrix
```



```
"""
```

Karena fungsi *matrix* menggunakan modul jadi untuk kode fungsi di bawah ini  
→dicomment.

Dan untuk code modul dengan fungsi di bawah ini adalah sama.

```
def createMatrix(rows,coloumn) :
    resultMatrix = []
    print("\n")

    for i in range(rows) :
        row = []
        print("\nBaris ",i)
        for j in range(coloumn) :
            row.append(int(input("=> Matrix [%d,%d] = " % (i,j) )))
        resultMatrix.append(row)

    return resultMatrix

def addMatrix(matrix1,matrix2) :
    checkSize = len(matrix1) == len(matrix2) and len(matrix1[0]) ==
    →len(matrix2[0])
    resultMatrix = []

    if checkSize :
        for i in range(len(matrix1)) :
            row = []
            for j in range(len(matrix1[0])) :
                row.append(matrix1[i][j]+matrix2[i][j])
            resultMatrix.append(row)

        return resultMatrix

    else :
        return "Ukuran matrix tidak sama"

def mulMatrix(matrix1,matrix2) :
    checkSize = len(matrix1[0]) == len(matrix2)
    resultMatrix = []

    if checkSize :
        for i in range(len(matrix1)) :
            tmpResult = []
            for j in range(len(matrix2[0])) :
```

```

        total = 0
        for k in range(len(matrix1)) :
            total+= matrix1[i][k]*matrix2[k][j]
            tmpResult.append(total)
        resultMatrix.append(tmpResult)
    return resultMatrix

else :
    return "Tidak Memenuhi"

"""

matrix1 = matrix.createMatrix(2,3)
matrix2 = matrix.createMatrix(2,3)
matrix3 = matrix.createMatrix(2,2)

jumlah1 = matrix.addMatrix(matrix1,matrix2)
jumlah2 = matrix.addMatrix(matrix1,matrix3)

kali1    = matrix.mulMatrix(matrix1,matrix2)
kali2    = matrix.mulMatrix(matrix1,matrix3)
kali3    = matrix.mulMatrix(matrix3,matrix2)

print("""

Matrix1 = %s
Matrix2 = %s
Matrix3 = %s

Matrix1 + Matrix2 = %s
Matrix1 + Matrix3 = %s

Matrix1 * Matrix2 = %s
Matrix1 * Matrix3 = %s
Matrix3 * Matrix2 = %s

""" %
(
    matrix1,matrix2,matrix3,
    jumlah1,jumlah2,
    kali1,kali2,kali3
)

)

```

```
Baris 0
=> Matrix [0,0] = 3
=> Matrix [0,1] = 1
=> Matrix [0,2] = 5
```

```
Baris 1
=> Matrix [1,0] = 9
=> Matrix [1,1] = 1
=> Matrix [1,2] = 4
```

```
Baris 0
=> Matrix [0,0] = 1
=> Matrix [0,1] = 4
=> Matrix [0,2] = 7
```

```
Baris 1
=> Matrix [1,0] = 8
=> Matrix [1,1] = 0
=> Matrix [1,2] = 2
```

```
Baris 0
=> Matrix [0,0] = 2
=> Matrix [0,1] = 1
```

```
Baris 1
=> Matrix [1,0] = 4
=> Matrix [1,1] = 8
```

```
Matrix1 = [[3, 1, 5], [9, 1, 4]]
Matrix2 = [[1, 4, 7], [8, 0, 2]]
Matrix3 = [[2, 1], [4, 8]]
```

```
Matrix1 + Matrix2 = [[4, 5, 12], [17, 1, 6]]
Matrix1 + Matrix3 = Ukuran matrix tidak sama
```

```
Matrix1 * Matrix2 = Tidak Memenuhi
Matrix1 * Matrix3 = Tidak Memenuhi
Matrix3 * Matrix2 = [[10, 8, 16], [68, 16, 44]]
```