


<p>Nama: Dave Ryano F.M</p> <p>NIM: 064002300039</p>	 <p>Praktikum Algoritma & Pemrograman</p>	<p>MODUL 8</p> <p>Nama Dosen: Ratna Shofiati, S.Kom, M. Kom</p>
<p>Hari/Tanggal: Kamis, 16 November 2023</p>		<p>Nama Asisten Labratorium: 1. Yuda Hadi Prasetyo – 065002100004 2. Muhammad Hasan Husein – 065002100009</p>

Fungsi Rekursif (Recursive Function)

1. Teori Singkat

Fungsi Rekursif

Fungsi rekursi adalah fungsi yang memanggil dirinya sendiri secara berulang. Jadi di dalam tubuh fungsi yang dideklarasikan kita memanggil fungsi itu sendiri. Rekursif ini sebenarnya merupakan sebuah perulangan di dalam sebuah program. Namun, perulangan rekursif ini sangat berbeda dengan perulangan pada umumnya, seperti while dan for. Walaupun fungsinya sama yaitu untuk melakukan perulangan atau looping. Letak perbedaannya adalah dari cara kerjanya. Jika 'for' dan 'while' merupakan sebuah perulangan yang menggunakan sebuah kondisi atau Boolean (true/false), maka pada rekursif ini terjadi pada sebuah fungsi atau metode yang memanggil dirinya sendiri. Dari penjelasan tersebut dapat kita katakan sebagai perulangan yang memanggil dirinya sendiri untuk melakukan sebuah perulangan.

Fungsi rekursif dapat menyelesaikan beberapa persoalan seperti perhitungan bilangan fibbonaci dan faktorial.

$\text{faktorial}(5) = 5 * \text{faktorial}(4)$
 $\text{faktorial}(4) = 4 * \text{faktorial}(3)$
 $\text{faktorial}(3) = 3 * \text{faktorial}(2)$
 $\text{faktorial}(2) = 2 * \text{faktorial}(1)$
 $\text{faktorial}(1) = 1$

Maka $\text{faktorial}(5) = 5 * 4 * 3 * 2 * 1$, akan menghasilkan 120



Source Code

```
#Fungsi Rekursif
def rekursif(angka):
    if angka > 0:
        print (angka)
        angka = angka - 1
        rekursif(angka)
    else:
        print(angka)

masukkan = int(input("masukkan angka: "))
rekursif(masukkan)
```

Output

```
masukkan angka: 5
5
4
3
2
1
0
```



2. Alat dan Bahan

Hardware : Laptop/PC

Software : Spyder (Anaconda Python)

3. Elemen Kompetensi

a. Latihan pertama

Buatlah sebuah fungsi penjumlahan berurut menggunakan konsep rekursif (tanpa for/while). User perlu memasukkan angka awal dan bilangan-bilangan yang ingin ditambahkan seperti pada berikut:

Masukkan Jumlah: 3 (inputan user)

Masukkan angka ke-1: 1 (inputan user)

Masukkan angka ke-2: 2 (inputan user)

Masukkan angka ke-3: 3 (inputan user)

Contoh output

```
Masukkan Jumlah: 3
Masukkan bilangan ke-1: 1
Masukkan bilangan ke-2: 2
Masukkan bilangan ke-3: 3
Hasil dari penjumlahan adalah: 6
```

Source Code

```
Created on Thu Nov 16 19:10:43 2023

@author: Oaan115
"""

def penjumlahan_urut(x):
    if x == 0:
        return 0
    else:
        angka = float(input(f"Masukkan angkanya ke-{x}: "))
        return angka + penjumlahan_urut(x - 1)

jumlah_input = int(input("Masukkan Jumlahnya: "))
hasil_penjumlahan = penjumlahan_urut(jumlah_input)
print(f"Hasil penjumlahannya: {hasil_penjumlahan}")
```



Output

```
C:\Users\Oaan115\Documents\Python Scripts>python Rekursif2.py
Masukkan Jumlahnya: 2
Masukkan angkanya ke-2: 20
Masukkan angkanya ke-1: 30
Hasil penjumlahannya: 50.0

C:\Users\Oaan115\Documents\Python Scripts>python Rekursif2.py
Masukkan Jumlahnya: 5
Masukkan angkanya ke-5: 10
Masukkan angkanya ke-4: 20
Masukkan angkanya ke-3: 30
Masukkan angkanya ke-2: 40
Masukkan angkanya ke-1: 50
Hasil penjumlahannya: 150.0

C:\Users\Oaan115\Documents\Python Scripts>python Rekursif2.py
Masukkan Jumlahnya: 10
Masukkan angkanya ke-10: 1
Masukkan angkanya ke-9: 2
Masukkan angkanya ke-8: 3
Masukkan angkanya ke-7: 4
Masukkan angkanya ke-6: 5
Masukkan angkanya ke-5: 6
Masukkan angkanya ke-4: 7
Masukkan angkanya ke-3: 8
Masukkan angkanya ke-2: 9
Masukkan angkanya ke-1: 10
Hasil penjumlahannya: 55.0
```

b. Latihan Kedua

Buatlah sebuah fungsi perpangkatan menggunakan konsep rekursif (tanpa for/while). User hanya perlu memasukkan base number (angka awal) dan power (pangkatnya). Hasil akhir berupa perhitungan perpangkatannya.

Contoh output

```
Ini merupakan program pemangkatan negatif dan positif, tekan enter untuk berhenti
Masukkan Angka: 2
Masukkan Pangkatnya: 2
Hasilnya adalah: 4

Ini merupakan program pemangkatan negatif dan positif, tekan enter untuk berhenti
Masukkan Angka:
Program Selesai
```

Source Code

```
Created on Thu Nov 16 19:10:43 2023

@author: Oaan115
"""

def penjumlahan_urut(x):
    if x == 0:
        return 0
    else:
        angka = float(input(f"Masukkan angkanya ke-{x}: "))
        return angka + penjumlahan_urut(x - 1)

jumlah_input = int(input("Masukkan Jumlahnya: "))
hasil_penjumlahan = penjumlahan_urut(jumlah_input)
print(f"Hasil penjumlahannya: {hasil_penjumlahan}")
```



Output

```
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Oaan115\Documents\Python Scripts>python Pangkat.py
Masukkan angka: 22
Masukkan pangkatnya: 2
Hasil perpangkatan: 484

C:\Users\Oaan115\Documents\Python Scripts>python Pangkat.py
Masukkan angka: 45
Masukkan pangkatnya: 100
Hasil perpangkatan: 2095324917039863304313251555826661356467173660585553096278749223514308239748113453774362345525549271
530430449123445790860832162183502447305727400816977024078369140625

C:\Users\Oaan115\Documents\Python Scripts>python Pangkat.py
Masukkan angka: 2
Masukkan pangkatnya: 2
Hasil perpangkatan: 4
```

4. File Praktikum

Github Repository:

5. Soal Latihan

Soal:

1. Sebutkan dan jelaskan perbedaan secara teknis antara perulangan for/while dengan perulangan fungsi rekursif?
2. Deskripsikan serta narasikan jalannya alur source code program yang sebelumnya telah kalian buat pada Elemen Kompetensi Latihan Kedua!

Jawaban:

1. Struktur dan cara kerjanya

for/while loop : Menggunakan struktur perulangan yang terdiri dari suatu blok kode yang dieksekusi secara berulang berdasarkan kondisi yang diberikan.

rekursi : Fungsi rekursif memanggil dirinya sendiri, menciptakan iterasi melalui serangkaian pemanggilan fungsi

Kinerja

for/while loop : lebih efisien untuk iterasi sederhana karena overhead pemanggilan fungsi rekursif dapat menyebabkan peningkatan konsumsi memori dan waktu eksekusi.

rekursi : Dapat memiliki overhead ekstra karena setiap panggilan rekursif menambahkan stack frame ke dalam memori.

2. Dengan menggunakan rumus def masukan penjumlahan urut $x = \text{angka}$, jika $x (\text{angka}) = 0$ maka hasilnya 0 dan else $\text{angka} = \text{float masukan angkanya nya ke-}\{x\}$ maka returnnya angka di jumlahkan ke penjumlahan urut $(x - 1)$.



6. Kesimpulan

- Dalam pengerjaan program dengan bahasa pemrograman Python, kita harus benar-benar teliti dalam menginputkan suatu fungsi untuk menampilkan suatu keluaran pada layar dengan sesuai.
- Kita dapat mengetahui Fungsi rekursif dalam algoritma memungkinkan suatu fungsi untuk memanggil dirinya sendiri, memecah permasalahan menjadi submasalah yang lebih kecil. dalam kata lain rekursif sama dengan perulangan namun cara kerja perulangan rekursif ini sangat berbeda dengan perulangan lainnya yang sebelumnya kita pelajari.

7. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	✓	
2.	Latihan Kedua	✓	

8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	15 Menit	Menarik
2.	Latihan Kedua	15 Menit	Menarik

Keterangan:

- Menarik
- Baik
- Cukup
- Kurang

