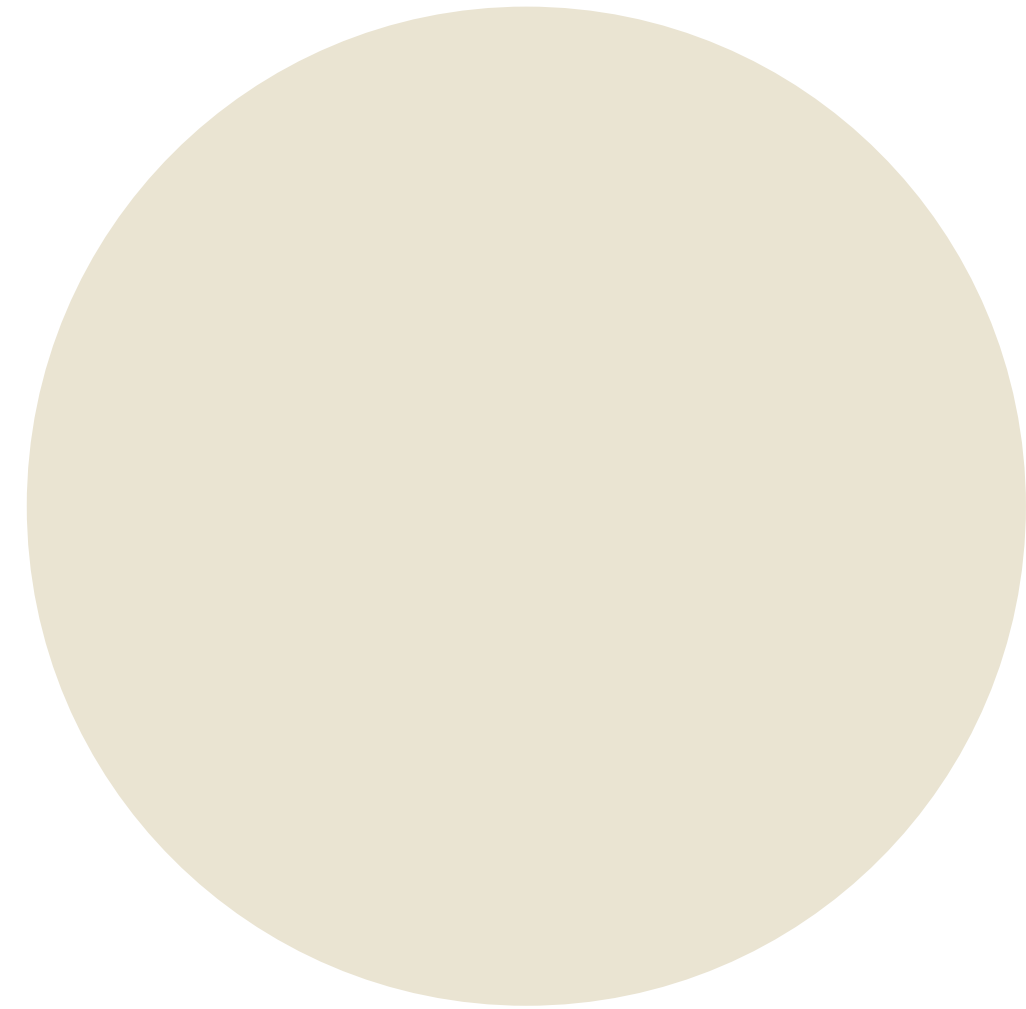




# ASA 3

● REKURSI



# REKURSI

Rekursi adalah teknik dalam pemrograman di mana suatu fungsi memanggil dirinya sendiri untuk menyelesaikan masalah yang lebih kecil hingga mencapai kondisi dasar (base case).

# BASIS DAN REKURENS

## Basis

Kondisi untuk menghentikan fungsi rekursi.

pada bagian ini terdiri dari fungsi yang terdefinisi secara eksplisit contoh :

```
if (n == 1) then return 1
```

(basis mirip dengan state pada repeat until yang berfungsi untuk menghentikan perulangan)

## Rekurens

rekurens merupakan bagian pada fungsi rekursi yang memanggil dirinya sendiri.

pada rekurens Berisi kaidah untuk menemukan nilai fungsi pada suatu input dari nilai-nilai lainnya pada input yang lebih kecil contoh :

```
else : return  $f(n-1) * n$ 
```

# FUNGSI FAKTORIAL



```
1  def faktorial(n):  
2      if n == 0:  
3          return 1  
4      else:  
5          return n * faktorial(n - 1)  
6  
7  print(faktorial(4))
```

Program tersebut merupakan fungsi rekursif untuk mencari  $n!$

Basis pada fungsi tersebut adalah

```
if n == 0:  
    return 1
```

Rekurens pada fungsi tersebut adalah

```
else:  
    return n * faktorial(n - 1)
```

# ALUR PEMANGGILAN FUNGSI FAKTORIAL

Sesuai definisi:

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

Sesuai definisi lagi, karena  
 $\text{faktorial}(3) = \text{faktorial}(2) * 3$ , maka:

$$\begin{aligned} \text{faktorial}(4) &= \text{faktorial}(3) * 4 \\ &= (\text{faktorial}(2) * 3) * 4 \end{aligned}$$

Apabila kita teruskan, kita dapatkan:

$$\begin{aligned} \text{faktorial}(4) &= \text{faktorial}(3) * 4 \\ &= (\text{faktorial}(2) * 3) * 4 \\ &= ((\text{faktorial}(1) * 2) * 3) * 4 \\ &= ((1 * 2) * 3) * 4 \\ &= (2 * 3) * 4 \\ &= 6 * 4 \\ &= 24 \end{aligned}$$

Ilustrasi lainnya adalah sebagai berikut:

$$\begin{aligned} \text{faktorial}(4) &= \text{faktorial}(3) * 4 \\ &= \begin{array}{|c|c|c|} \hline & \text{faktorial}(2) & * 3 \\ \hline \end{array} * 4 \\ &= \begin{array}{|c|c|c|c|} \hline & & \text{faktorial}(1) * 2 & * 3 \\ \hline \end{array} * 4 \\ &= \begin{array}{|c|c|c|c|c|} \hline & & & 1 & * 2 \\ \hline \end{array} * 3 * 4 \end{aligned}$$

# REKURSI PADA PROCEDURE

umumnya rekursi dilakukan pada fungsi namun dalam pemrograman rekursi juga dapat dilakukan pada procedure

contoh dalam program menggambar segitiga dengan '\*'

\*

\*\*

\*\*\*

```
1  def segitiga(n):
2      if n == 0:
3          print()
4      else:
5          segitiga(n-1)
6          for i in range(n):
7              print('*', end='')
8          print()
9
10 segitiga(3)
```

# MENGGAMBAR PEGUNUNGAN

gambarlah pegunungan dengan aturan :

- diawali dengan pegunungan tingkat  $N-1$ ,
- dilanjutkan dengan satu baris berisikan  $N$  buah karakter  $*$ ,
- dan diakhiri dengan pegunungan tingkat  $N-1$ .

contoh pegunungan tingkat 1 :

\*

contoh pegunungan tingkat 2 :

\*

\*\*

\*

contoh pegunungan tingkat 3 :

\*

\*\*

\*

\*\*\*

\*

\*\*

\*

# MENGGAMBAR PEGUNUNGAN

```
1  def gunung(n):
2      if n == 0:
3          print()
4
5      else :
6          gunung(n-1)
7          for i in range(n):
8              print('*', end='')
9          gunung(n-1)
10
11 gunung(3)
```

Pada program tersebut basisnya adalah

```
if n == 0:
    print()
```

rekurensnya adalah :

```
else :
    gunung(n-1)
    for i in range(n):
        print('*', end='')
    gunung(n-1)
```



# ALUR PEMANGGILAN PROCEDUR MENGGAMBAR GUNUNG

definisi :

```
gunung(3)
```

lalu kita uraikan gunung(3) :

```
gunung(3) = gunung(2)
           '***'
           gunung(2)
```

lalu kita uraikan gunung(2) :

```
gunung(3) = gunung(1)
           '**'
           gunung(1)
           '***'
           gunung(1)
           '**'
           gunung(1)
```

# ALUR PEMANGGILAN PROCEDUR MENGGAMBAR GUNUNG

Sehingga menghasilkan :

```
gunung(3) = ' * '  
            ' ** '  
            ' * '  
            ' *** '  
            ' * '  
            ' ** '  
            '* '*
```

# THANK YOU



● FOR YOUR NICE ATTENTION

---

link canva: [https://www.canva.com/design/DAGgzEmWcr4/w1SLJkahg0kE\\_aaVctk0bw/edit?  
utm\\_content=DAGgzEmWcr4&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAGgzEmWcr4/w1SLJkahg0kE_aaVctk0bw/edit?utm_content=DAGgzEmWcr4&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)