

REKURSIF

1. Tujuan Instruksional Umum

- a. Mahasiswa dapat melakukan perancangan aplikasi menggunakan algoritma rekursif.
- b. Mahasiswa mampu melakukan analisis pada algoritma rekursif yang dibuat.
- c. Mahasiswa mampu mengimplementasikan algoritma rekursif pada sebuah aplikasi secara tepat dan efisien.

2. Tujuan Instruksional Khusus

- a. Mahasiswa dapat menjelaskan mengenai algoritma rekursif.
- b. Mahasiswa dapat membuat dan mendeklarasikan Abstraksi Tipe Data rekursif.
- c. Mahasiswa mampu menerapkan rekursif.

Pengertian Algoritma Rekursif

Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri. Fungsi ini akan terus berjalan sampai kondisi berhenti terpenuhi, oleh karena itu dalam sebuah fungsi rekursif perlu terdapat 2 blok penting, yaitu blok yang menjadi titik berhenti dari sebuah proses rekursi dan blok yang memanggil dirinya sendiri.

Contoh konsep penggunaan Rekursif

Masalah : Memotong Roti tawar tipis-tipis sampai habis

Algoritma :

1. Jika roti sudah habis atau potongannya sudah paling tipis maka pemotongan roti selesai.
2. Jika roti masih bisa dipotong, potong tipis dari tepi roti tersebut, lalu lakukan prosedur 1 dan 2 untuk sisa potongannya.

Contoh Fungsi Rekursif

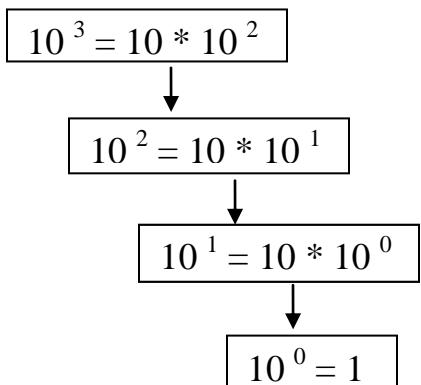
- a. Fungsi pangkat
- b. Faktorial
- c. Fibonancy
- d. Menara Hanoi

Fungsi Pangkat

Menghitung 10^n dengan menggunakan konsep rekursif.

Secara Notasi pemrograman dapat ditulis :

Contoh :



Faktorial

$$0! = 1$$

$$N! = N \times (N-1)! \text{ Untuk } N > 0$$

Secara notasi pemrograman dapat ditulis sebagai :

Contoh :

$$\text{FAKT}(5) = 5 * \text{FAKT}(4)$$

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

$$\text{FAKT}(3) = 3 * \text{FAKT}(2)$$

$$\text{FAKT}(2) = 2 * \text{FAKT}(1)$$

$$\text{FAKT}(1) = 1 * \text{FAKT}(0)$$

Misal :

hitung $5!$, maka dapat dilakukan secara rekursif dgn cara :

$$5! = 5 * 4!$$

Secara rekursif nilai dari $4!$ dapat dihitung kembali dgn $4 * 3!$, sehingga $5!$

Menjadi : $5! = 5 * 4 * 3!$

Secara rekursif nilai dari $3!$ dapat dihitung kembali dgn $3 * 2!$, sehingga $5!$

Menjadi : $5! = 5 * 4 * 3 * 2!$

Secara rekursif nilai dari $2!$ dapat dihitung kembali dgn $2 * 1$, sehingga $5! = 5 * 4 * 3 * 2 * 1 = 120$.

Berikut ini adalah fungsi faktorial rekursif dari sebuah program.

```
def fibonacci(n):
    # Basis: jika n = 0 atau 1, langsung kembalikan n
    if n <= 1:
        return n
    else:
        # Rekurens:  $F_n = F_{n-1} + F_{n-2}$ 
        return fibonacci(n - 1) + fibonacci(n - 2)
```

Fibonancy

Deret Fibonancy : 0,1,1,2,3,5,8,13,.....

Secara notasi pemrograman dapat ditulis sebagai :

$$\text{Fibo}(1) = 0 \text{ & } \text{Fibo}(2) = 1 \dots \dots \dots \quad (1)$$

$$\text{Fibo}(N) = \text{Fibo}(N-1) + \text{Fibo}(N-2) \dots \dots \dots \quad (2)$$

Contoh :

$$\text{Fibo}(5) = \text{Fibo}(4) + \text{Fibo}(3)$$

$$\text{Fibo}(4) = \text{Fibo}(3) + \text{Fibo}(2)$$

$$\text{Fibo}(3) = \text{Fibo}(2) + \text{Fibo}(1)$$

Latihan 1

Akan dibuat program yang akan menerima sebuah bilangan bulat ***bil*** dan mencetak bilangan fibonaci yang ke-***bil***. Untuk menentukan bilangan fibonacci, dibuat sebuah fungsi dengan nama fibonacci.

Ketiklah program berikut :

```
def fibonacci(i):
    """
    Mengembalikan bilangan ke-i dari deret Fibonacci.
    """

    if i == 2 or i <= 1:
        return 0
    else:
        return fibonacci(i - 1) + fibonacci(i - 2)
```

```
def main():
    while True:
        bil = int(input("Masukkan bilangan: "))
        print(f"Fibonacci({bil}) = {fibonacci(bil)}")

        lagi = input("Cari lagi (y/t)? ")
        if lagi.lower() != 'y':
            break

if __name__ == "__main__":
    main()
```

Latihan 2

Membuat program untuk menghitung sebuah factorial. Ketiklah list program berikut ini :

```
def fact_rec(n):
    """
    Menghitung nilai faktorial dari n secara rekursif.
    """

    if n < 0:
        return 0
    elif n == 0 or n == 1:
        return 1
    else:
        return n * fact_rec(n - 1)

def main():
    fac = int(input("Masukkan berapa faktorial: "))
    print(f"Hasil faktorial dari {fac} adalah: {fact_rec(fac)}")

if __name__ == "__main__":
    main()
```

Tugas

1. Buat program untuk menggambar segitiga sama kaki siku-siku terbalik, segitiga sama kaki siku-siku tidak terbalik, dan segitiga sama kaki (piramid) terbalik. Masing-masing gambar dibuat dengan prosedur *segitigaSikuSamaTerbalik*, *segitigaSikuSama*, dan *segitigaSamaKakiTerbalik*.

Buat file baru, ketik program berikut, dan simpan dengan nama "karakter.py"

```
def cetakKarakter(cc, n):
    """
    Mencetak karakter 'cc' sebanyak 'n' kali secara horizontal.
    """

    if n > 0:
        print(cc, end='') # cetak tanpa pindah baris
        cetakKarakter(cc, n - 1)

def segitigaSikuSamaTerbalik(n):
    """
    Mencetak segitiga siku-siku sama kaki terbalik dengan tinggi n.
    """

    if n > 0:
        cetakKarakter('*', n)
        print()
        segitigaSikuSamaTerbalik(n - 1)
```

Buat file baru, ketik program berikut, kompilasi dan jalankan.

```
def segitigaSikuSama(n):
    """
    Mencetak segitiga siku-siku sama kaki normal (tegak) dengan tinggi n.
    """

    if n > 0:
        segitigaSikuSama(n - 1)
        print()
        cetakKarakter('*', n)

def segitigaSamaKakiTerbalik(tinggi, posisiAwal):
    """
    Mencetak segitiga sama kaki terbalik dengan tinggi tertentu.
    posisiAwal mengatur jarak spasi awal.
    """

    if tinggi > 0:
        cetakKarakter(' ', posisiAwal - 1)
        cetakKarakter('*', 2 * tinggi - 1)
        print()
        segitigaSamaKakiTerbalik(tinggi - 1, posisiAwal + 1)

def main():
    while True:
        n = int(input("Masukkan sebuah bilangan: "))

        print("\nSegitiga Siku-siku Sama Kaki Terbalik:")
        segitigaSikuSamaTerbalik(n)

        print("\nSegitiga Siku-siku Sama Kaki Tegak:")
        segitigaSikuSama(n)

        print("\nSegitiga Sama Kaki Terbalik:")
        segitigaSamaKakiTerbalik(n, 1)

        lagi = input("\nBuat segitiga lagi (y/t)? ")
        if lagi.lower() != 'y':
            break

    if __name__ == "__main__":
        main()
```

2. Buatlah fungsi untuk membalik suatu bilangan dengan cara rekursif. Sebagai contoh, bilangan 1234 ditampilkan menjadi 4321! (Take Home)