

BAB 7

SUBPROGRAM (METHOD DI
JAVA) DAN REKURSIF

Definisi

- Subprogram merupakan program bagian dengan blok terpisah dan diluar program utama, dan akan dipanggil pada program utama jika subprogram itu diperlukan untuk dijalankan.

Macam Subprogram

Di bahasa pemrograman yang lain biasa disebut function tetapi di java yang berbasis OOP subprogram di sebut sebagai sebuah method (perilaku object)

digunakan karena

- Merupakan konsep program modular yaitu memecah-mecah program yang rumit menjadi bagian yang lebih sederhana
- Untuk pengolahan yang sering dilakukan berulang-ulang, cukup dituliskan sekali saja dalam sub program dan dapat dipanggil atau digunakan sewaktu-waktu bila diperlukan.

Istilah-istilah di method

- Method disebut juga Subprogram/function
- Parameter adalah nilai yang digunakan method untuk diolah didalam method
`Math.pow(x,y);` // x dan y adalah parameter
- Return adalah pengembalian nilai yang diperoleh dari hasil pengolahan pada method
`double hasil = Math.pow(x,y);`
// hasil adalah nilai olahan method



Contoh syntax method

Syntax tanpa paramater tanpa return

```
public class cetakgaris {  
    public static void main(String[] args) {  
        garis();  
        public static void garis()  
        {System.out.println("-----"  
            ");}  
    }  
}
```

Tipe data void (tanpa return)

Tanpa deklarasi variabel untuk parameter

Contoh dengan parameter tanpa return

```
public class cetaknama {  
    public static void main(String[] args) {  
        nama("aku");  
        public static void nama(String x)  
        {System.out.println("nama saya :"+x);}  
    }  
}
```

Tipe data void (tanpa return)



Terdapat deklarasi variabel parameter



Dengan parameter dan dengan pengembalian nilai (return)

```
public class jumlahangka {  
    public static void main(String[] args) {  
        Int a=3,b=4;  
        System.out.println("jumlahnya :"+jumlah(a,b));}  
    public static int jumlah(int x, int y)  
    {int z=x+y;  
    return z;}  
}
```

Terdapat deklarasi parameter

Tipe data return

Terdapat keyword return untuk variabel yang sesuai tipe datanya

Batasan Variabel

- *Variabel dalam suatu method bersifat Lokal* artinya bahwa nilai yang terdapat didalam suatu modul program hanya dapat digunakan pada modul yang bersangkutan sehingga tidak dapat digunakan pada bagian program lain. (di java variabel dibatasi oleh blok)

Pemanggilan Parameter

Pemanggilan parameter pada prosedur terdiri dari 2 yaitu :

- Parameter Secara Nilai (By Value)

Bersifat satu arah, jika terjadi perubahan pada subprogram, maka tidak akan mempengaruhi program utama.(terjadi pada variabel primitif)

- Parameter Secara Acuan (By Reference)

Bersifat dua arah, jika terjadi perubahan pada subprogram, maka program utamanya akan ikut berubah. (misal terjadi pada variabel array)

Contoh dengan parameter by value

```
public class ParByValue {  
    public static void main(String[] args) {  
        int x=1;  
        ubahnilai(x);  
        System.out.println("nilai x :"+x);  
        public static void ubahnilai(int a)  
        {a++;}  
    }  
}
```

Nilai x tetap seperti di main
program



Contoh dengan parameter by reference

```
public class ParByRef {  
    public static void main(String[] args) {  
        int [] x={1,2}; ubahnilai(x);  
        System.out.println("nilai x[0] :"+x[0]);  
        System.out.println("nilai x[1] :"+x[1]);  
    }  
    public static void ubahnilai(int [] a)  
    {  
        a[0]=a[0]*10;  
        a[1]=a[1]*10;  
    }  
}
```

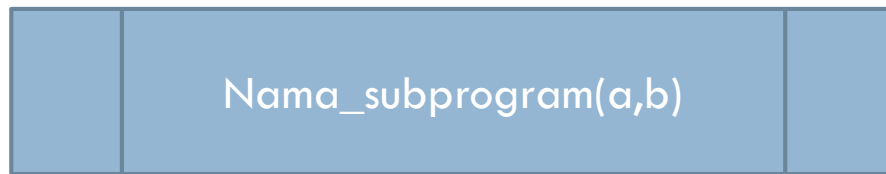
Nilai berubah mengikuti method (method menjadi refensi)



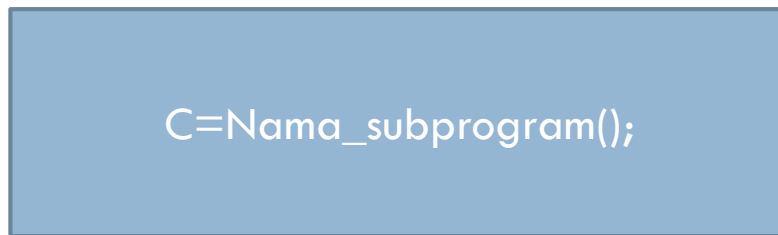


Algoritma Method

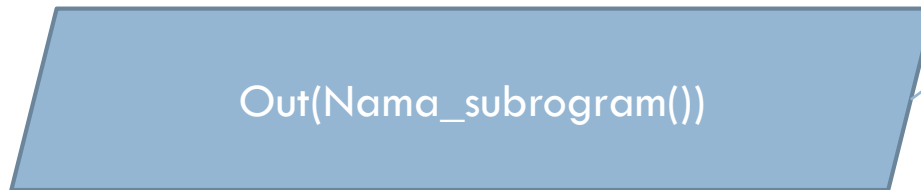
Lambang subprogram di flowcart



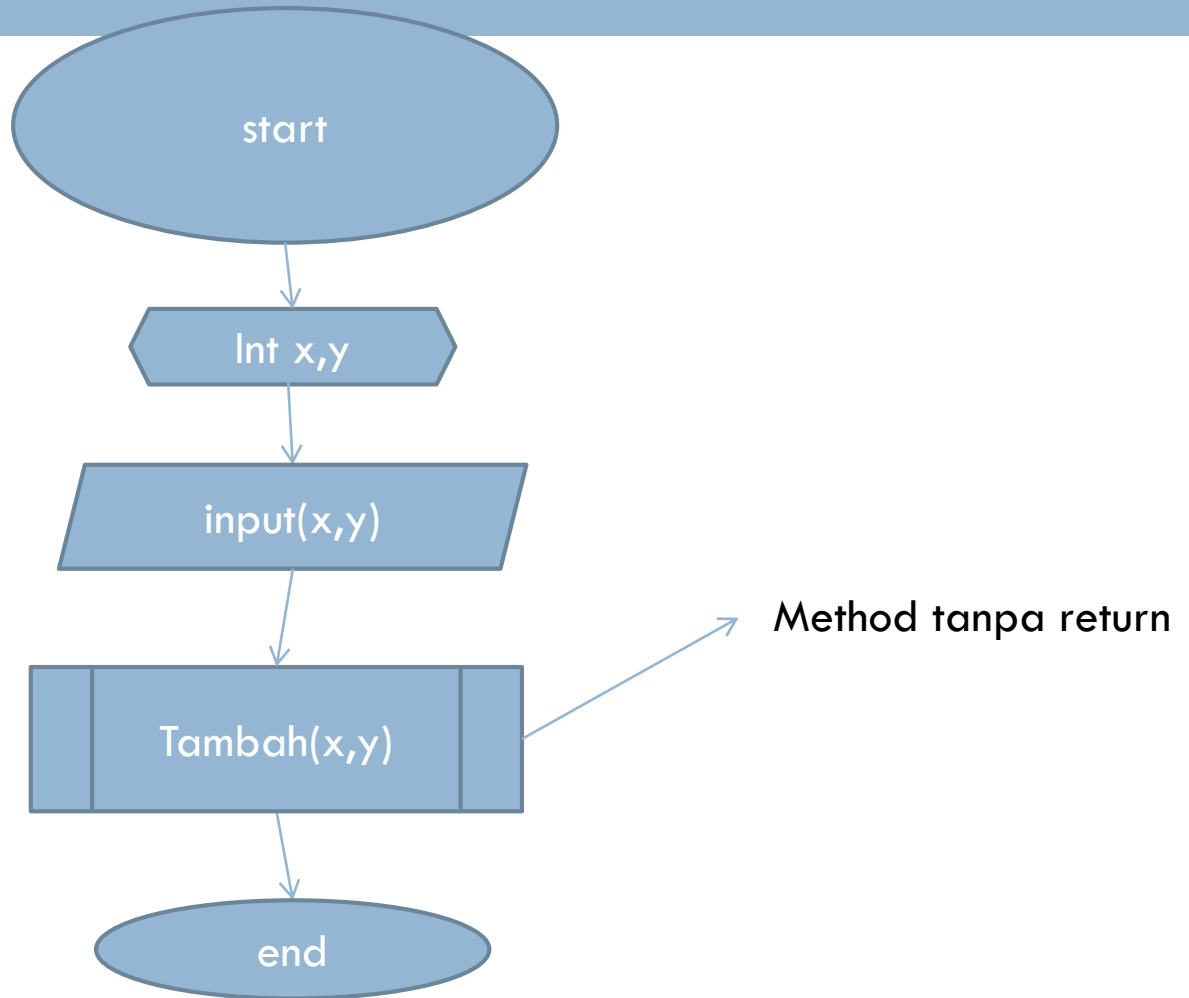
Dengan parameter
tanpa return



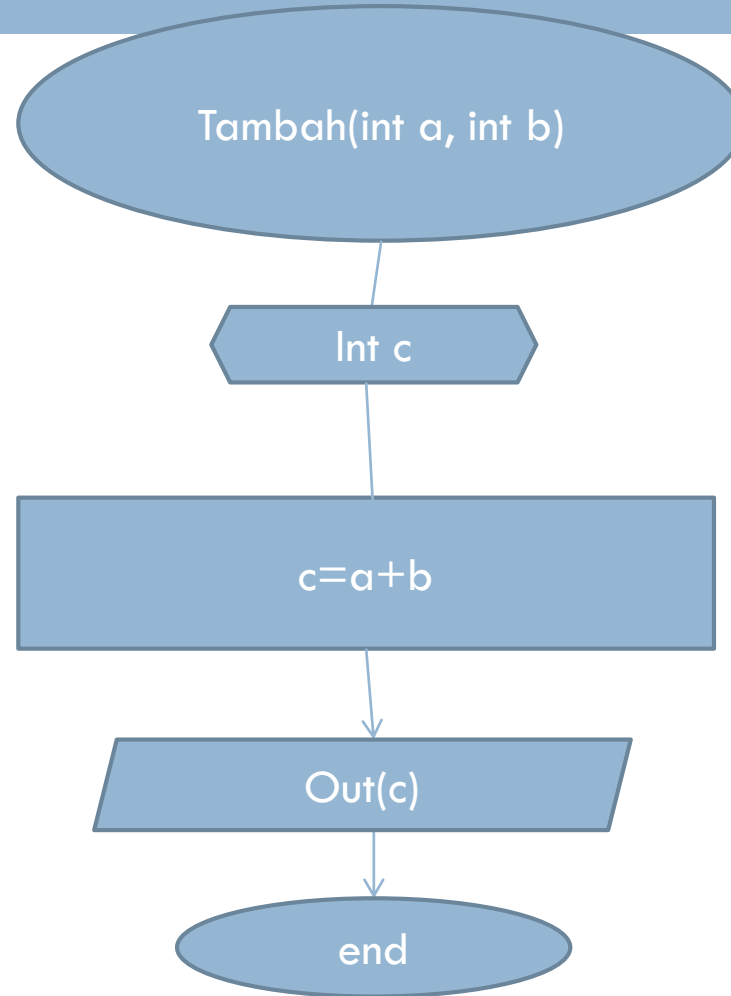
Tanpa parameter
dengan return



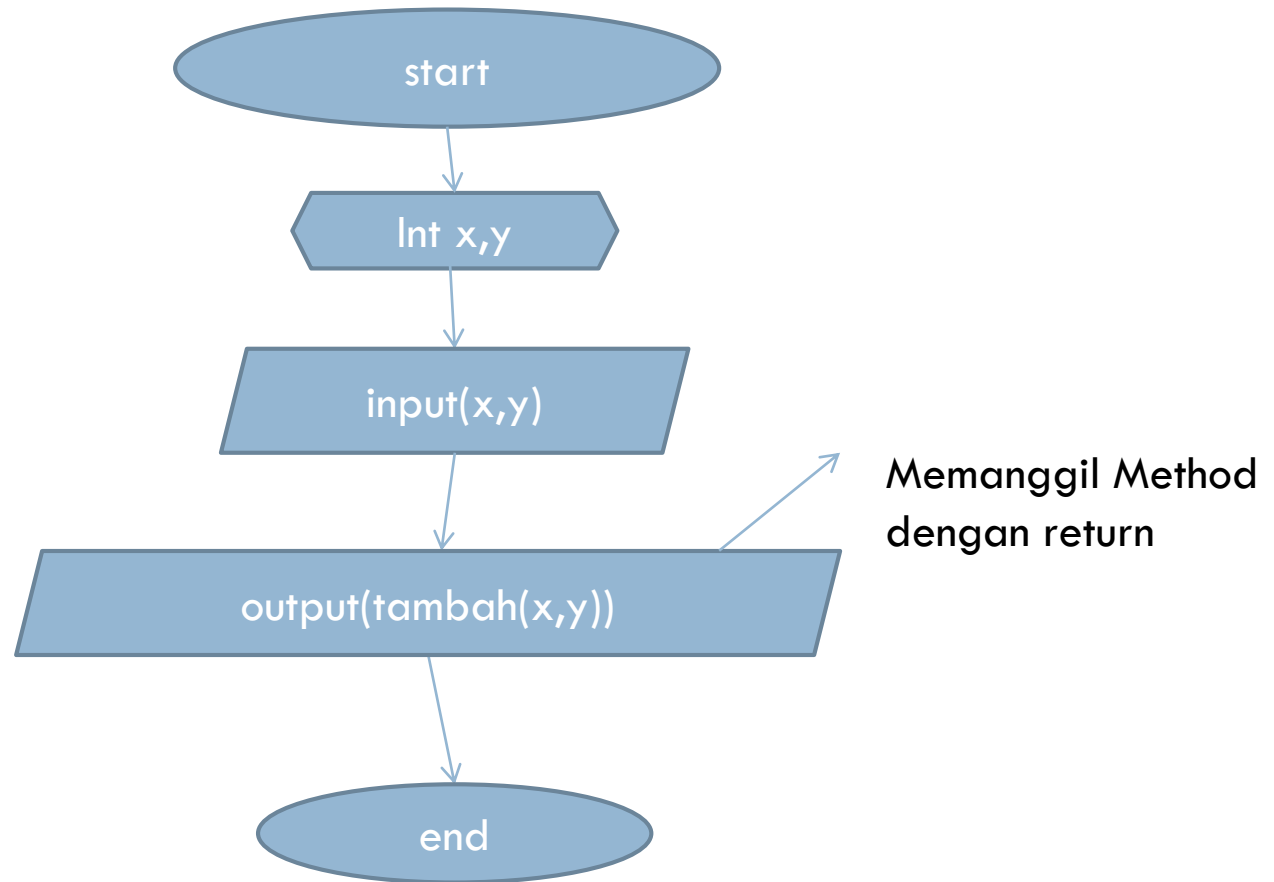
Flowchar pertambahan



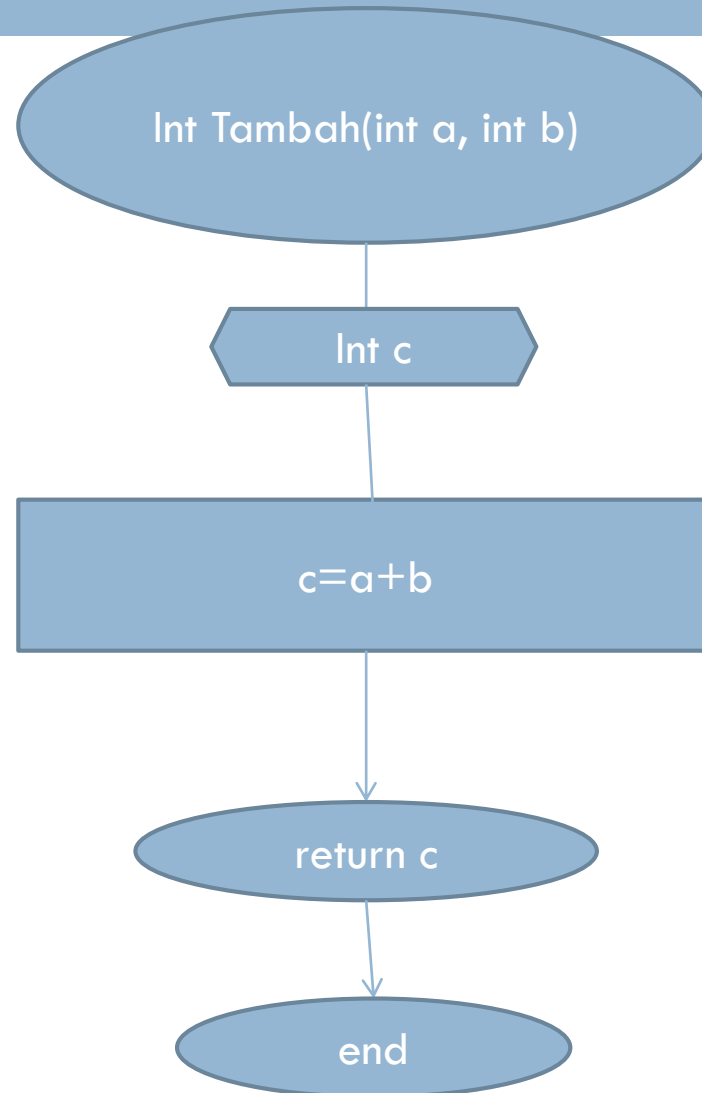
Contoh flowchart subprogram dengan parameter tanpa return



Flowchar pertambahan versi 2



Contoh flowchart subprogram dengan parameter dengan return



Latihan 1

Membuat flowchart dan program dengan method dengan return untuk:

1. Mencari Absolut dari nilai bil bulat
2. Mencari maksimal dari 3 data integer
3. Menentukan jenis segitiga dari 3 sisi yang diinputkan (segitiga sama sisi, sama kaki atau segitiga sembarang)
4. Mencari n faktorial
5. Menukar 2 data
6. Mencari n faktorial dengan rekursif

Fungsi Rekursif

21

- Fungsi yang berisi definisi dirinya sendiri atau pada matematika rekursif adalah sebuah fungsi yang didalamnya ada fungsi itu sendiri.
- Prosesnya terjadi secara berulang-ulang (tapi tidak menggunakan perintah perulangan hanya dengan jika maka
- Yang perlu diperhatikan adalah “stopping role”(membuat fungsi itu berhenti menghitung atau memproses)

Plus – Minus Fungsi Rekursif

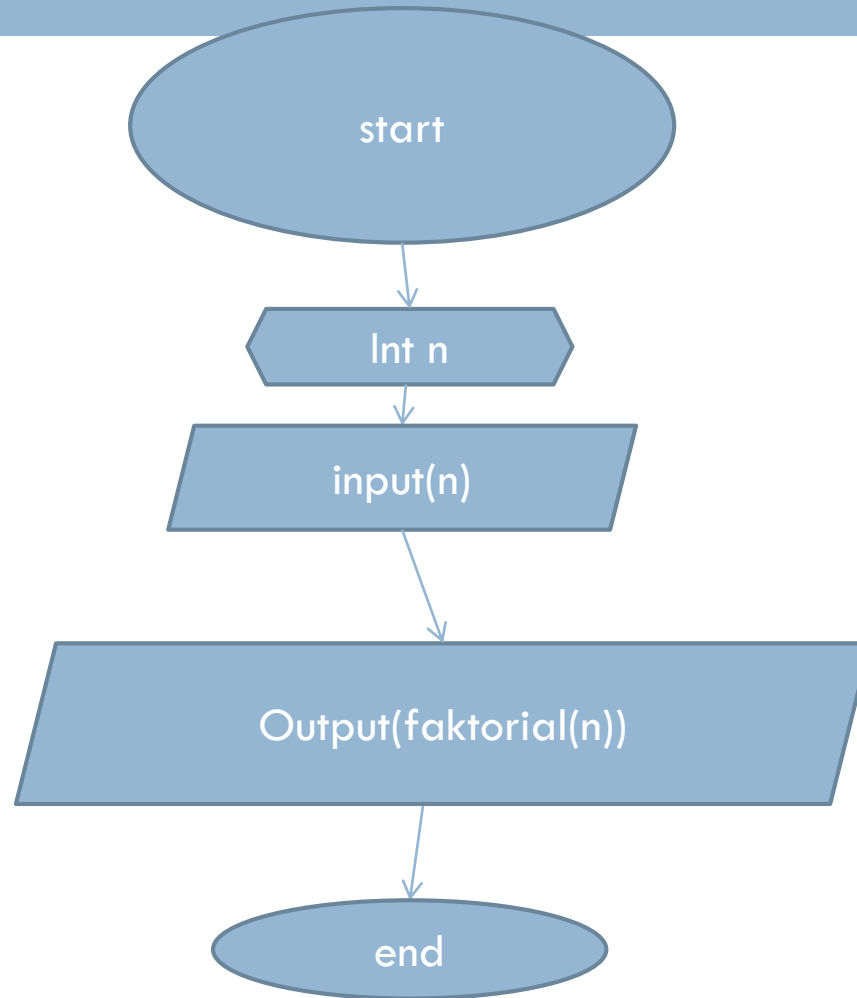
- Karena program lebih singkat dan ada beberapa kasus yang lebih mudah menggunakan fungsi yang rekursif
- Memakan memori yang lebih besar, karena setiap kali bagian dirinya dipanggil, dibutuhkan sejumlah ruang memori tambahan.
- Mengorbankan efisiensi dan kecepatan
- Problem: rekursi seringkali tidak bisa “berhenti” sehingga memori akan terpakai habis dan program bisa hang.
- Saran: jika memang bisa diselesaikan dengan iteratif, gunakanlah iteratif!

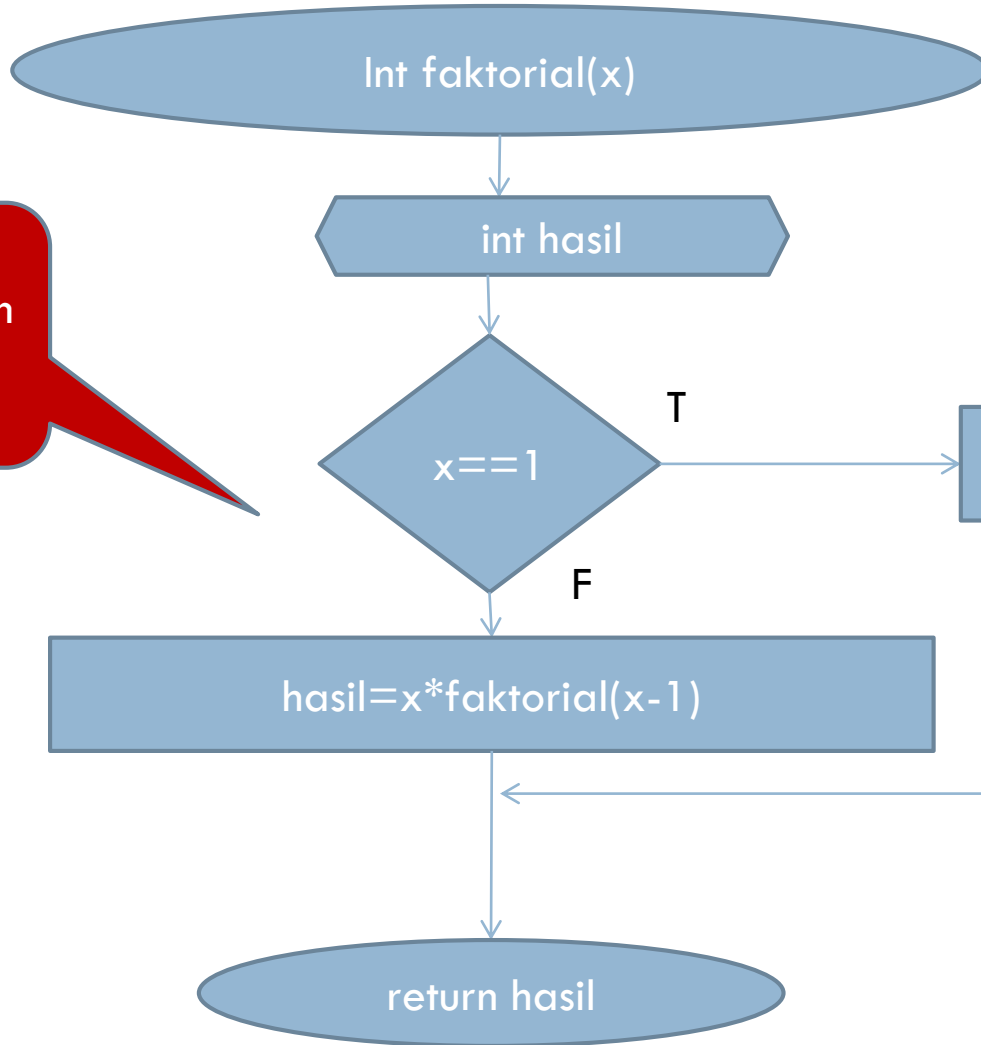
Contoh metode rekursif untuk kasus perulangan

Cara lain selain dengan looping untuk menyelesaikan permasalahan faktorial adalah dengan cara rekursi, dimana n faktorial adalah :

$$\text{Faktorial}(n) = n * \text{Faktorial}(n-1)$$

Flowchart faktorial





Percabangan bukan perulangan

Latihan 2

Buat rancangan program dan program dengan menggunakan fungsi return tanpa rekursi dan fungsi rekursif (jadi satu program untuk 1 kasus) untuk :

1. Menghitung x pangkat y
2. Mencari jumlah suku (S_n) deret Aritmatika dengan inputan a, b , dan n
3. Mencari jumlah suku (S_n) dari deret geometri dengan inputan a, r dan n
4. Menentukan deret ke- n dari deret Fibonacci (1,1,2,3,5,8,13,...)