

# **STRUKTUR DASAR ALGORITMA**

# **STRUKTUR DASAR ALGORITMA**

## **Pernyataan**

Sebuah algoritma merupakan deskripsi langkah-langkah pelaksanaan suatu proses. Setiap langkah di dalam algoritma dinyatakan dalam sebuah pernyataan atau istilah lainnya instruksi. Sebuah pernyataan berisi aksi *(action)* yang dilakukan.

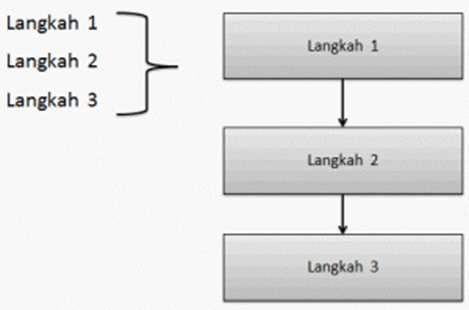
## **Kontruksi Dasar**

Algoritma berisi langkah-langkah penyelesaian suatu masalah. Langkah-langkah tersebut dapat berupa runtunan aksi, pemilihan aksi, dan pengulangan aksi. Ketiga langkah tersebut membentuk konstruksi suatu algoritma, jadi sebuah algoritma dapat dibangun dari ketiga buah konstruksi dasar yaitu:

1. **Runtunan *(sequence),***
2. **Pemilihan *(selection)*,** dan
3. **Pengulangan *(repetition)*.**

## **Runtutan**

Runtunan terdiri dari satu atau lebih pernyataan dengan setiap pernyataan dikerjakan secara berurutan sesuai urutannya. Urutan instruksi menentukan keadaan akhir algoritma, bilamana urutannya diubah maka hasil akhirnya juga berubah.



## **Pemilihan**

Instruksi dikerjakan jika memenuhi kondisi tertentu seperti kasus *traffic light,* jika lampu berwarna merah maka kendaraan harus berhenti

Pernyataan:

Jika lampu *traffic light* berwarna merah, maka berhenti

Pernyataan pemilihan: if kondisi then aksi

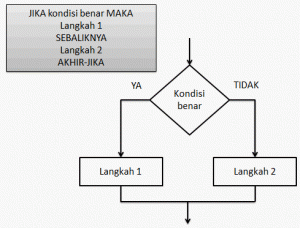
if *kondisi* then

*aksi 1*

else

*aksi 2*

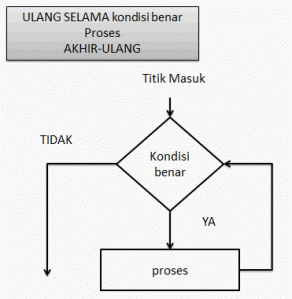
\*else artinya kalau tidak



## **Pengulangan**

Salah satu kelebihan komputer adalah kemampuannya untuk mengerjakan pekerjaan yang sama berulang kali tanpa lelah. Ada 3 notasi pengulangan di algoritma :

1. **for**
2. **repeat**
3. **while**



## **Implementasi**

1. **Runtunan**

Hitunglah jumlah dari 3 buah bilangan bulat !

Buatlah algoritma dengan flowchart & bahasa naturalnya dari masalah di atas.

Output : tulis (jumlah)

Jawaban :

Bahasa Natural

1. Mulai
2. Masukan A,B dan C
3. Jumlahkan nilai A,B dan C. hasilnya simpan di variable jumlah
4. Cetak Jumlah
5. Selesai

**Flowchart**



1. **Pemilihan**

Program untuk mengecek apakah sebuah bilangan termasuk bilangan genap atau bukan. Jika termasuk bilangan genap tampilkan pesan.

**PROGRAM** Bilangan\_genap;

uses crt;

**var** bil:integer;

**begin**

clrscr;

write('Masukan bilangan :');

readln(bil);

if bil mod 2=0 then

**begin**

**writeln**('bilangan genap');

**writeln**('pengecekan bilangan selesai');

**end;**

**readln;**

**end.**

1. **Pengulangan**

Program untuk mengecek apakah sebuah bilangan termasuk bilangan genap atau bilangan ganjil. Jika termasuk bilangan genap, tampilkan pesan ’bilangan genap’. Jika termasuk bilangan ganjil tampilkan pesan’bilangan ganjil’.

**PROGRAM** genap\_ganjil;

uses crt;

**var** bil:integer;

**begin**

clrscr;

**write**('masukan bilangan :'); readln(bil);

if bil mod 2 = 0 then

**writeln**('bilangan genap')

else

**writeln**('bilangan ganjil');

**readln;**

**end.**

## **LATIHAN**

1. **Runtunan**

Buatlah algoritma (flowchart dan Bahasa Natura ) untuk mencari hasil dari dua buah bilangan !

Output : tulis (hasil).

1. **Pemilihan**

Buatlah Program untuk mengecek apakah sebuah bilangan termasuk bilangan ganjil atau bukan. Jika termasuk bilangan ganjil tampilkan pesan.

1. **Pengulangan**

Buatlah Program untuk mengecek apakah sebuah bilangan habis dibagi 2 atau bilangan tidak habis dibagi 2. Jika termasuk bilangan habis dibagi 2, tampilkan pesan **Bilangan Habis**. Jika termasuk bilangan tidak habis dibagi 2 tampilkan pesan **Bilangan Masih Tersisa**.