

STRUKTUR KONTROL

SEQUENCE

- Merupakan struktur kontrol paling sederhana.
- Berisi deretan instruksi dan/atau aksi yang apabila dieksekusi maka akan berjalan sesuai dengan urutan penulisannya.
- Contoh dari sequence adalah algoritma pindah isi bejana :

a = susu;

b = kopi;

c = a;

a = b;

b = c;

STRUKTUR KONTROL

- Terdapat struktur kontrol yang memungkinkan user merubah alur program saat dieksekusi
- Terdapat tiga macam, yaitu :
 1. Struktur kontrol pemilihan (if-else, switch-case) yang digunakan untuk memilih blok kode yang akan dieksekusi
 2. Struktur kontrol pengulangan (while-do, for-to-do) yang digunakan untuk mengeksekusi blok tertentu berulang kali.
 3. percabangan (break, continue, return) yang digunakan untuk mengatur arah dari aliran program

STRUKTUR KONTROL PEMILIHAN

- Adalah pernyataan yang mengijinkan ueser untuk memilih dan mengeksekusi blok kode spesifik dan mengabaikan blok kode yang lain
- Blok kode yang dimaksud adalah sebuah operator bersyarat dengan kondisi yang telah ditentukan.
- Memiliki dua macam cara yaitu dengan menggunakan if-then-else atau dengan menggunakan switch-case

STATEMENT IF-THEN

- Pernyataan if akan menentukan sebuah pernyataan (blok kode) yang akan dieksekusi jika dan hanya jika persyaratan bernilai benar(true).
- Bentuk dari pernyataan if

```
if (boolean_expression) {  
    statement;  
}
```
- Dimana, boolean_expression adalah sebuah kondisi yang diwakilkan dengan pernyataan logika (true/false)
- Contoh :

```
int iNilai = 75;  
if (iNilai > 60){  
    output("Selamat Nilai anda bagus");  
}
```

STATEMENT IF-THEN-ELSE

- Merupakan perkembangan dari statement if-then. Digunakan apabila kita ingin mengeksekusi beberapa pernyataan dengan kondisi true dan pernyataan yang lain dengan kondisi false.
- Bentuk statement if-else:

```
if (boolean_expresion){  
    statement;//kondisi true  
}  
else{  
    statement;//kondisi false  
}
```

STATEMENT IF-THEN-ELSE .. 2

- Contoh :

```
int iNilai = 75;
```

```
if (iNilai > 55){
```

```
    output("Selamat anda lulus");
```

```
} else {
```

```
    output("maaf anda belum lulus");
```

```
}
```

STATEMENT IF-THEN-ELSE .. 3

Merupakan perkembangan dari statement if-then-else. Digunakan apabila kita ingin mengeksekusi beberapa pernyataan dengan kondisi true dan pernyataan yang lain dengan kondisi false.

Bentuk statement if-else:

```
    if (boolean_expresion){
        statement;//kondisi true
    }else if (Boolean_expression 2){
        statement;//kondisi true ke 2
    } else if (Boolean_expression 3){
        statement;//kondisi true ke 2
    }
    ...
} else {
    Statement; //kondisi false
}
```


STATEMENT IF-THEN-ELSE .. 4

- Contoh :

```
int iNilai = 75;
if (iNilai > 80){
    output("Nilai anda A");
} else if (iNilai > 65){
    output("Nilai anda B");
} else if (iNilai > 55) {
    output("nilai anda C");
} else {
    output("nilai anda E");
}
```

STATEMENT SWITCH-CASE

- program akan menjalankan statement sesuai dengan “nilai” yang didapat.
- Pengkondisian dengan switch – case adalah pengontrol alur program dengan menggunakan operasi integer (byte, short, dan int)
- Bentuk dari statement switch_case :

```
Switch (variable){  
    case nilai_variable 1 : kondisi 1;break;  
    case nilai_variable 2 : kondisi 2;break;  
    default : kondisi selain 1 dan 2;break;  
}
```

STATEMENT SWITCH-CASE

- contoh:
- `Int bulan = 1;`
`Switch (bulan){`
 `case 1 : output ("Januari"); break;`
 `case 2 : output ("Februari"); break;`
 `case 3 : output ("Maret"); break;`
 .
 .
 .
 `default : output ("bulan hanya sampai 12 belum ada penambahan") ; break;`
`}`

OPERATOR PEMBANDING

- Misal int x = 20, String a = "alpro"

| Operator | Deskripsi | Contoh | Hasil |
|----------|----------------------|-------------------|-------|
| == | Sama dengan | x == 20 | True |
| != | Tidak sama dengan | x != 20 | False |
| > | Lebih besar | x > 20; | False |
| < | Lebih kecil | x < 100; | True |
| >= | Lebih besar dari | x >= 20; | True |
| <= | Lebih kecil dari | x <= 20; | True |
| equals | Sama dengan (String) | a.equals("alpro") | True |