

BAHASA PEMROGRAMAN *BASIC*

OBJEKTIF:

1. Mahasiswa dapat memahami kondisi perulangan `if then-end if`.
 2. Mahasiswa dapat memahami kondisi perulangan `if then-else-end if`.
 3. Mahasiswa dapat memahami kondisi perulangan `if then-elseif-end if`.
 4. Mahasiswa dapat memahami kondisi perulangan `select case-end select`.
 5. Mahasiswa mampu mengenali perbedaan penggunaan kondisi perulangan `if then-end if`, `if then-else-end if`, `if then-elseif-end if`, dan `select case-end select`.
-

2.1 Struktur Kendali Kondisi

Struktur kendali memegang peranan yang sangat penting dalam pembuatan program. Tanpa adanya struktur kendali, kita hanya dapat membuat program sederhana, yang prosesnya hanya dijalankan dari baris pertama sampai selesai. Pada kasus tertentu, ada kalanya baris program tertentu tidak boleh dilaksanakan apabila kondisi tidak terpenuhi, atau baris-baris tertentu harus dibuang sampai suatu kondisi terpenuhi. Dengan penggunaan kondisi struktur kendali, program baru dapat menentukan jalur-jalur proses yang harus dikerjakan oleh komputer, berdasarkan logika tertentu dengan perintah *test* pilihan.

ALUR PROGRAM



Struktur kendali digunakan untuk mengatur aliran program berdasarkan kondisi yang terpenuhi. Terdapat 2 (dua) jenis untuk struktur kendali kondisi, yaitu struktur kendali percabangan (struktur kondisional) dan perulangan (*looping*). Struktur percabangan digunakan untuk menjalankan perintah jika kondisinya terpenuhi.

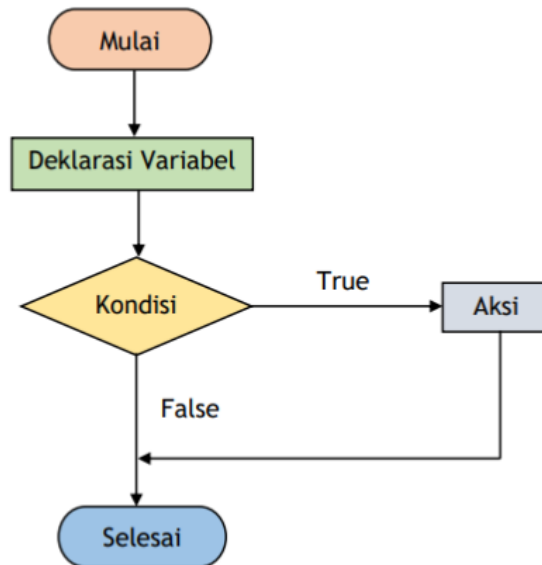
2.1.1 Kondisi perulangan `if then-end if` (Pilih satu)

Kondisi `if then-end` ini berfungsi untuk memilih apakah kondisi (ekspresi logika) memenuhi syarat untuk melakukan *statement-statement* yang berada dalam blok `if then-end if`.

Bentuk umum `if then-end if` :

```
IF KONDISI THEN  
Statement  
END IF
```

Berikut ini merupakan alur program dari `if then-end if` :



Saat program dimulai, kita akan mendeklarasikan variabel yang digunakan. Kemudian, akan masuk ke dalam kondisi. Kondisi di sini menggunakan tipe data boolean yang akan bernilai `true` jika kondisinya benar, dan `false` jika kondisinya salah. Jika kondisi `true` atau benar, maka akan menjalankan aksi, dan program pun akan selesai. Namun, jika saat pengecekan kondisi hasilnya `false` atau salah, maka program akan selesai.

Contoh program:

```
Dim nilai1 As Double  
Dim nilai2 As Double  
Dim nilai3 As Double  
Dim rata_rata As Double  
  
INPUT "Nilai ujian 1: "; nilai1  
INPUT "Nilai ujian 2: "; nilai2  
INPUT "Nilai ujian 3: "; nilai3  
  
rata_rata = (nilai1 + nilai2 + nilai3) / 3.0  
  
Print "Nilai rata-rata ketiga ujiannya adalah: ";rata_rata  
  
If rata_rata > 89 THEN  
    Print "Nilai yang sangat bagus!"  
End If
```

Output program:

```
Nilai ujian 1: 90
Nilai ujian 2: 89.5
Nilai ujian 3: 88
Nilai rata-rata ketiga ujiannya adalah: 89.16666666666667
Nilai yang sangat bagus!
```

atau

```
Nilai ujian 1: 50.6
Nilai ujian 2: 60.4
Nilai ujian 3: 90.9
Nilai rata-rata ketiga ujiannya adalah: 67.3
```

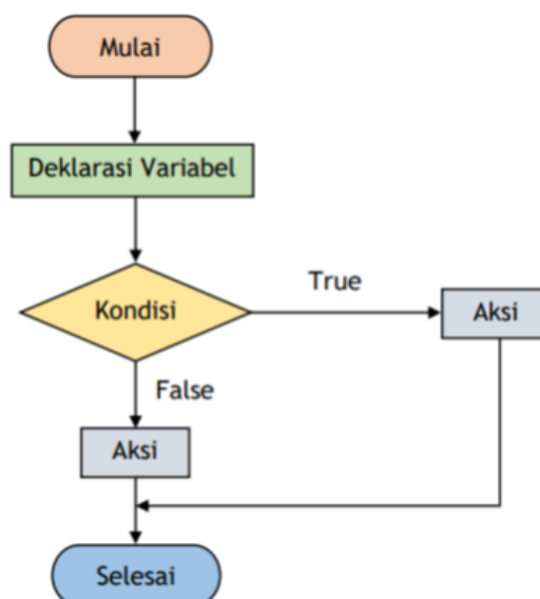
2.1.2 Kondisi perulangan `if then-else-end if` (Pilih antara dua)

Kondisi `if then-else-end if` berfungsi untuk menentukan statement mana yang akan dijalankan oleh komputer yang sesuai dengan ekspresi logika, yang menghasilkan nilai `true` atau benar.

Bentuk umum `if then-else-end if` :

```
IF KONDISI THEN
Statement1
ELSE Statement2
END IF
```

Berikut ini merupakan alur program dari `if then-else-end if` :



Saat program dimulai, kita akan mendeklarasikan variabel yang digunakan. Kemudian, akan masuk ke dalam kondisi. Kondisi di sini menggunakan tipe data boolean yang akan bernilai `true` jika kondisinya benar, dan `false` jika kondisinya salah. Jika kondisi `true` atau benar, maka akan menjalankan aksi, dan program pun akan selesai. Namun, saat pengecekan kondisi, jika kondisi salah `false` atau salah, maka akan menjalankan aksi dari kondisi `false` atau salah, dan program pun akan selesai.

Jadi, perbedaannya jika kondisi `false` atau salah di sini tetap akan menjalankan aksi. Namun, aksi yang akan dijalankan berbeda dengan aksi pada kondisi `true` atau benar.

Contoh program:

```
Dim vaksin as String
Dim usia as String

Input "Masukkan usia anda: "; usia
Input "Apakah anda sudah vaksin? (ya/tidak) ";vaksin

If vaksin = "ya" and usia >= "18" then
    Print "Anda dapat melakukan penerbangan"
Else
    Print "Anda tidak dapat melakukan penerbangan"
End if
```

Output program:

```
Masukkan usia anda: 22
Anda dapat melakukan penerbangan
```

atau

```
Masukkan usia anda: 15
Anda tidak dapat melakukan penerbangan
```

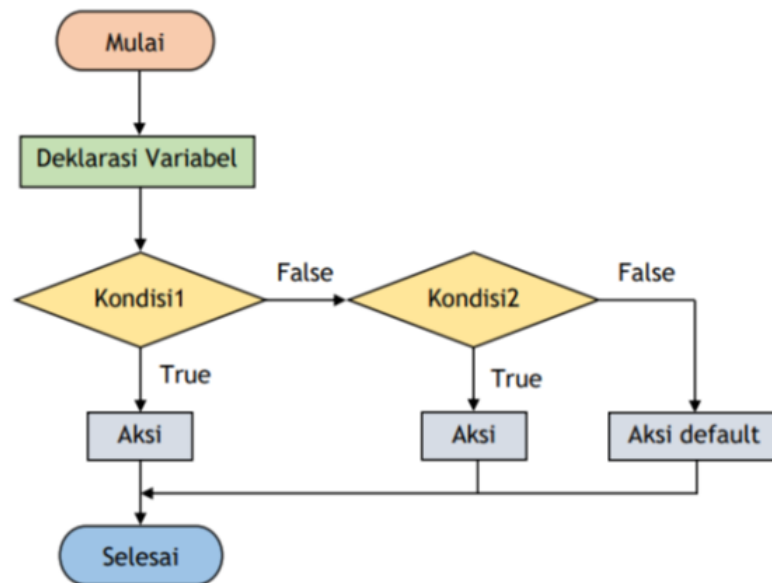
2.1.3 Kondisi perulangan `if then-elseif-end if` (Pilih antara banyak)

Kondisi perulangan `if then-elseif-end if` berfungsi untuk menentukan statement mana yang akan dijalankan dari banyak pilihan yang sesuai dengan ekspresi logika.

Bentuk umum `if then-elseif-end if` :

```
IF KONDISI THEN
Statement1
ELSEIF KONDISI
THEN
Statement2
ELSE
Statement-n
END IF
```

Berikut ini merupakan alur program dari `if then-elseif-end if` :



Saat program dimulai, kita akan mendeklarasikan variabel yang digunakan. Lalu, melakukan pengecekan kondisi1. Jika `true` atau benar, maka akan menjalankan aksi, dan program pun selesai. Namun, jika saat pengecekan kondisi1, dan kondisinya `false` atau salah, maka akan mengecek kondisi2. Lalu, jika kondisi2 ini bernilai `true` atau benar, maka aksi akan dijalankan dan program pun selesai. Tetapi, jika kondisi1 dan kondisi2 salah, maka akan menjalankan aksi `default`, dan program akan selesai.

Contoh program 1:

```
Dim total as integer

Input "Masukkan total harga belanja anda: ";total

If total <= 100000 Then
    Print "Maaf, Anda tidak mendapatkan potongan harga"
Elseif total <= 200000 Then
    Print "Anda mendapatkan potongan harga 10%"
Elseif total <= 300000 Then
    Print "Anda mendapatkan potongan harga 20%"
Else
    Print "Anda mendapatkan potongan harga 30%"
End if
```

Output program 1:

```
Masukkan total harga belanja anda: 57000
Maaf, Anda tidak mendapatkan potongan harga
```

atau

```
Masukkan total harga belanja anda: 500000
Anda mendapatkan potongan harga 30%
```

Contoh program 2:

```
REM Program Menampilkan Nilai

'Deklarasi variabel'
DIM skor AS Double

'Minta skor dari pengguna'
INPUT "Masukkan Skor nilai Anda: ";skor

'Statement if untuk seleksi nilai'
If skor >= 89 then
    Print "Nilai Anda: A"
Elseif skor >= 69 then
    Print "Nilai Anda: B"
Elseif skor >= 49 then
    Print "Nilai Anda: C"
Elseif skor >= 19 then
    Print "Nilai Anda: D"
Else
    Print "Nilai Anda: E"
End If
```

Output program 2:

```
Masukkan skor nilai Anda: 56
Nilai Anda: C
```

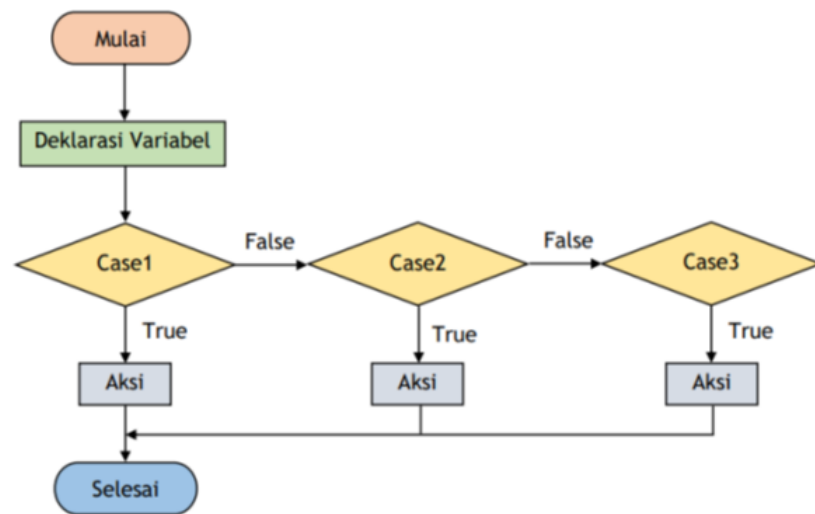
2.1.4 Kondisi perulangan `select case-end select` (Pilih antara banyak)

Kondisi perulangan `select case-end select` ini sama seperti `if then-elseif-end if` yang berfungsi untuk menentukan statement mana yang akan dijalankan dari banyak pilihan yang sesuai dengan ekspresi logika. Perbedaannya, jika `select case-end select` ini, menggunakan `case` tergantung dari berapa pilihannya, jika terdapat 3 pilihan, maka terdapat 3 `case`.

Bentuk umum *select case-end select* :

```
SELECT CASE nama_variable
CASE ekspresi1
    Statement1
CASE ekspresi2
    Statement2
CASE ELSE
    Statement3
END SELECT
```

Berikut ini merupakan alur program dari `select case-end select` :



Saat program dimulai, kita akan mendeklarasikan variabel yang digunakan. Perbedaan dari kondisi sebelumnya, di sini kita menggunakan `case`, jadi, pilihannya tergantung dari banyak `case` nya. Jika nilai yang diinput sesuai dengan case1, artinya nilainya adalah `true` atau benar, maka akan menjalankan aksi, dan program pun selesai. Tetapi, jika nilai yang diinput tidak sesuai dengan case1 yang artinya `false` atau salah, maka akan memeriksa case2. Lalu, jika nilai sesuai dengan case2 yang bernilai `true` atau benar, maka akan menjalankan aksi, dan program pun selesai. Tetapi, jika nilai tidak sesuai dengan case1 maupun case2, yang artinya case1 `false` atau salah, case2 `false` atau salah, maka akan menjalankan case3 dan menjalankan aksi, dan program pun selesai.

Contoh program 1:

```
Dim nama as string
Dim npm as string
Dim jurusan as string

Input "Masukkan nama anda : ";nama
input "Masukkan npm anda : ";npm

Print "Pilihan : "
Print "1. Teknik Informatika"
Print "2. Teknik Komputer"
Print "3. Teknik Elektro"

Input "Masukkan pilihan jurusan anda : ";jurusan
Print

Select case jurusan
Case "1"
    print nama ;" dengan npm ";npm; " memilih jurusan Teknik Informatika"
Case "2"
    print nama ;" dengan npm ";npm; " memilih jurusan Teknik Komputer"
Case "3"
    print nama ;" dengan npm ";npm; " memilih jurusan Teknik Elektro"
Case else
    print "Anda salah menginputkan pilihan jurusan"
```

```
End select
```

Output program 1:

Pilihan :

1. Teknik Informatika
2. Teknik Komputer
3. Teknik Elektro

Rizky Afif dengan npm 12345678 memilih jurusan Teknik Informatika

Contoh program 2:

```
REM Program Menu Luas Bangun Datar
```

```
'Deklarasi variabel'
```

```
DIM menu as Integer
```

```
DIM sisi as Double
```

```
DIM panjang as Double
```

```
DIM lebar as Double
```

```
DIM alas as Double
```

```
DIM tinggi as Double
```

```
DIM jarijari as double
```

```
DIM luas_segitiga as double
```

```
DIM luas_persegi as double
```

```
DIM luas_persegipanjang as double
```

```
DIM luas_lingkaran as double
```

```
'Tampilkan Menu Pilihan'
```

```
PRINT "MENU Mencari Luas Bangun Datar"
```

```
PRINT "=====
```

```
PRINT "1. Luas Segitiga"
```

```
PRINT "2. Luas Persegi"
```

```
PRINT "3. Luas Persegi Panjang"
```

```
PRINT "4. Luas Lingkaran"
```

```
PRINT "=====
```

```
INPUT "Masukkan pilihan Anda (ketik No. 1, 2, 3, atau 4): "; menu
```

```
'Statement Case untuk setiap pilihan menu'
```

```
Select Case menu
```

```
Case 1:
```

```
    INPUT "Masukkan alas segitiga (cm): "; alas
```

```
    INPUT "Masukkan tinggi segitiga (cm): "; tinggi
```

```
    luas_segitiga = (alas * tinggi) / 2
```

```
    PRINT "Luas segitiga: "; luas_segitiga; " cm"
```

```
Case 2:
```

```
    INPUT "Masukkan sisi persegi (cm): "; sisi
```

```
    luas_persegi = sisi * sisi
```

```
    PRINT "Luas persegi: "; luas_persegi; " cm"
```


Case 3:

```
INPUT "Masukkan panjang persegi panjang (cm): ";panjang
INPUT "Masukkan lebar persegi panjang (cm): ";lebar

luas_persegipanjang = panjang * lebar
PRINT "Luas persegi panjang: "; luas_persegipanjang;" cm"
```

Case 4:

```
INPUT "Masukkan jari-jari lingkaran (cm): ";jarijari

luas_lingkaran = 2 * 3.14 * jarijari
PRINT "Luas lingkaran: "; jarijari;" cm"
```

End Select

Output program 2:

MENU Mencari Luas Bangun Datar

=====

1. Luas Segitiga
2. Luas Persegi
3. Luas Persegi Panjang
4. Luas Lingkaran

=====

Masukkan pilihan Anda (ketik No. 1, 2, 3, atau 4): 1

Masukkan alas segitiga (cm): 5

Masukkan tinggi segitiga (cm): 10

Luas segitiga: 25 cm

REFERENSI

- [1] FreeBASIC. Diakses melalui <https://www.freebasic.net/>. Diakses pada 15 Oktober 2020.
- [2] documentation.help. *FreeBASIC Documentation*. Diakses melalui <https://documentation.help/FreeBASIC/KeyPgIfthen.html>. Diakses pada 15 Oktober 2020.