

PERTEMUAN 5
STATEMENT PEMILIHAN

Algoritma dan Pemrograman I

Teknik Informatika

UNIKOM

Analisis Satu Kasus

- Menggunakan konstruksi IF-THEN (jika-maka) dalam bentuk pernyataan :

```
if (kondisi)
   then
      pernyataan
endif
```

Contoh Masalah satu kasus :

- Buatlah algoritma untuk menentukan apakah suatu bilangan yang dimasukkan oleh user itu bilangan genap atau bilangan ganjil.

- Penyelesaian :

  Bilangan genap adalah bilangan yang habis dibagi dengan 2 (sisa pembagian = 0). Oleh karena itu, kita perlu membagi data masukan dengan 2. Jika data masukan habis dibagi 2, maka dapat ditulis bahwa bilangan tersebut bilangan genap, jika tidak maka akan ditulis bilangan tersebut bilangan ganjil

Analisis Dua kasus :

- Menggunakan konstruksi IF-THEN-ELSE (jika-maka-kalau tidak) dalam bentuk pernyataan :

```
if (kondisi)
   then
         pernyataan1
   else
         pernyataan2
endif
```

Contoh masalah analisis dua kasus :

Tulislah algoritma yang membaca sebuah bilangan bulat, lalu menuliskan pesan “genap” jika bilangan tersebut adalah genap atau “ganjil” jika bilangan tersebut adalah bilangan ganjil, dengan menggunakan analisis dua kasus.

•Penyelesaian :

Misalkan bilangan yang dibaca adalah bil. Hanya ada dua kemungkinan jenis untuk bil, yaitu bilangan genap atau bilangan ganjil. Bilangan genap adalah bilangan yang habis dibagi dengan 2 (sisa pembagian = 0), sedangkan bilangan ganjil bersisa 1 bila dibagi dengan 2. Contohnya, 4 adalah bilangan genap karena 4 mod 2 = 0, tapi 3 adalah bilangan ganjil karena 3 mod 2 =1.

Analisis Tiga Kasus atau Lebih :

Tiga Kasus :

```
if kondisi1 then
   pernyataan1
else
   if kondisi2 then
         pernyataan2
   else
         if kondisi3 then
               pernyataan3
         endif
   endif
endif
```

Empat Kasus :

```
if kondisi1 then
   pernyataan1
else
   if kondisi2 then
         pernyataan2
   else
         if kondisi3 then
               pernyataan3
         else
               if kondisi4 then
                     pernyataan4
               endif
         endif
   endif
endif
```

Struktur Depend On

Konstruksi Depend On adalah sebagai berikut :

Depend On (*ekspresi)*
*nilai*1 : *pernyataan*1
*nilai*2 : *pernyataan*2
*nilai*3 : *pernyataan*3
. . .
*nilai*n : *pernyataan*n
else : *pernyataan*x
EndDepend

Ekspresi adalah sembarang ekspresi (aritmatika atau boolean) atau variabel yang menghasilkan suatu nilai (konstanta).

Struktur Case

Konstruksi CASE adalah sebagai berikut :

```
case (ekspresi) of
    nilai1 : pernyataan1
    nilai2 : pernyataan2
    nilai3 : pernyataan3
             . . .
    nilain : pernyataannn
    else : pernyataanx
endcase
```

Ekspresi adalah sembarang ekspresi (aritmatika atau boolean) atau variabel yang menghasilkan suatu nilai (konstanta).

Konstruksi CASE yang ekivalen dengan konstruksi IF-THEN-ELSE

```
if (ekspresi = nilai1) then
        pernyataan1
else
        if (ekspresi = nilai2) then
                pernyataan2
        else
                if (ekspresi = nilai3) then
                        pernyataan3
                        . . .
                        if (ekspresi = nilain) then
                                pernyataann
                        else  { otherwise }
                                pernyataanx
                        endif
                endif
        endif
endif
```

Contoh Masalah :

- Buatlah algoritma yang membaca sebuah bilangan bulat yang nilainya terletak antara 1 sampai 4, lalu mencetak teks angka tersebut. Misalkan bila dibaca angka 1, maka tercetak tulisan “satu”, bila dibaca 2, maka tercetak di layar tulisan “dua”, demikian seterusnya. Jika angka yang dimasukkan selain 1 sampai 4, tuliskan pesan bahwa angka yang dimasukkan salah.

Penyelesaian :

```
ALGORITMA KonversiAngkaTeks
{ Mencetak kata untuk angka 1 sampai 4 }
Kamus :
        angka : integer   { angka yang dibaca }
Algoritma :
        input(angka)
        if angka = 1 then
            output(“satu”)
        else
            if angka = 2 then
                output(“dua”)
            else
                if angka = 3 then
                    output(‘tiga’)
                else
                    if angka = 4 then
                            output(‘empat’)
                    else
                            output(‘angka yang dimasukkan salah’)
                    endif
                endif
            endif
        endif
```

Dengan konstruksi CASE, algoritma untuk masalah di atas dapat dibuat menjadi lebih singkat sebagai berikut :

```
ALGORITMA KonversiAngkaTeks
{ Mencetak kata untuk angka 1 sampai 4 }
Kamus :
        angka : integer  { angka yang dibaca }
Algoritma :
        input(angka)
        case angka
            1 : output("satu")
            2 : output("dua")
            3 : output("tiga")
            4 : output("empat")
         else : output("angka yang dimasukkan salah")
        endcase
```

Tugas

Buatlah Algoritma untuk menu pilihan sebagai berikut :

Menu Pilihan

1.Menghitung Luas Persegipanjang

2.Menghitung Luas Segitiga

0. Keluar