


Pertemuan 2

Percabangan Sederhana
MK. Algoritma dan Struktur Data

Present: Ponco WP

Created: Bekti Wulandari, M.Pd.
TE KELAS B
2014



Definisi Percabangan

- ⊙ Percabangan adalah suatu perintah (pernyataan) yang memungkinkan suatu perintah (pernyataan) dieksekusi jika suatu kondisi terpenuhi atau tidak terpenuhi.
- ⊙ Jika suatu kondisi terpenuhi, maka perintah akan dilaksanakan.
- ⊙ Jika kondisi tidak terpenuhi, maka perintah yang lainnya yang dilaksanakan.

Definisi Percabangan

(lanjutan)

- ⊙ Percabangan (branching) di dalam pemrograman digunakan oleh komputer untuk menentukan langkah kerja instruksi.
- ⊙ Percabangan menggunakan operator kondisional yang akan menghasilkan nilai boolean (benar/true atau salah/false).
- ⊙ Jika nilai yang dihasilkan benar, maka perintah (instruksi) akan dilaksanakan, sedangkan jika salah, maka instruksi tidak akan dilaksanakan atau melaksanakan instruksi lainnya.

Macam Percabangan

1. Satu Kasus

- if kondisi then aksi1
- Notasi algoritma :
if kondisi-terpenuhi (true) then
 laksanakan_aksi
 endif
- kondisi berupa ekspresi yang menghasilkan true /false
- aksi berupa instruksi yang akan dilaksanakan jika kondisi yang dipasangkan dengan aksi yang bersangkutan bernilai benar. Bila kondisi bernilai salah, tidak ada pernyataan apapun yang dikerjakan

- Contoh
Mencetak pesan “bilangan genap” jika bilangan tersebut genap.

Macam Percabangan

(lanjutan 1)

2. Dua Kasus

- if kondisi then aksi1 else aksi2
- Notasi Algoritma
if kondisi-terpenuhi (true) then
 laksanakan_aksi
 else kondisi_tidak_terpenuhi (false)
 endif
- Digunakan untuk menguji sebuah kondisi dimana jika kondisi terpenuhi maka perintah yang telah ditentukan akan dijalankan, tetapi jika kondisi tidak terpenuhi maka perintah yang lain yang akan dijalankan.

- Contoh

Mencetak pesan “bilangan genap” jika bilangan tersebut bilangan genap, atau “bilangan ganjil” jika bilangan tersebut ganjil.

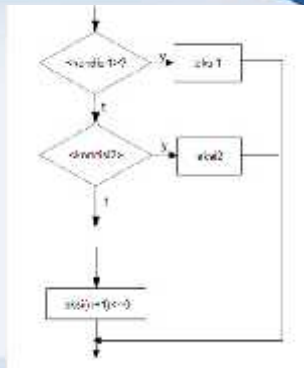
Macam Percabangan

(lanjutan 1)

3. Tiga Kasus atau Lebih

- if kondisi1 then
aksi1
else
if kondisi2 then
aksi2
else aksi3
endif
endif
- Hampir sama dengan bentuk percabangan kedua tetapi kondisi yang diuji lebih dari satu.

- Flowchart



- Contoh

Membaca temperatur air (T) pada tekanan normal (dalam satuan derajat celsius). Lalu menentukan apakah wujud air tersebut dalam keadaan padat ($T \leq 0$), cair ($0 < T < 100$), gas ($T \geq 100$)

Algoritma :

Program.....

Deklarasi

T :.....

Algoritma

Read (...)

if $T \leq 0$ then

Write ('.....')

else if (.....) and (.....) then

Write ('.....')

if $T \geq 100$ then

write('.....')

end if

end if

endif

Soal

1. Karyawan honorer di PT ABC digaji berdasarkan jumlah jam kerja selama satu minggu. Upah per jam Rp 2000,-. Bila jumlah jam kerja lebih besar dari 48 jam, maka sisanya dianggap sebagai jam lembur. Upah lembur Rp 3000,-/jam. Tulislah algoritma yang membaca nama pegawai jumlah jam kerja seorang karyawan selama satu minggu, lalu menentukan upah minggunya.
2. Menampilkan bilangan terbesar dari tiga bilangan yang dimasukkan!
3. Buatlah algoritma untuk menentukan apakah bilangan bulat yang dimasukkan tersebut bilangan positif, bilangan negatif, atau bilangan nol!

Soal

(lanjutan)

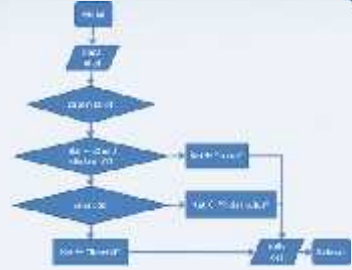
4. Buatlah algoritma yang membaca sebuah titik $P(x,y)$ di bidang kartesian, lalu menentukan di kuadran mana letak titik tersebut!
5. Mengurutkan tiga bilangan yang dimasukkan dari kecil ke besar.
6. Mengurutkan tiga bilangan yang dimasukkan dari kecil ke besar dimana bilangan yang dimasukkan tidak boleh ada yang sama.

Struktur CASE

- Digunakan untuk memilih jika terdapat lebih dari dua kondisi
- Case ekspresi of
 - nilai1: aksi1
 - nilai2: aksi2
 - nilai3: aksi3
 -
 - nilaiN:aksiN
 - otherwise:aksiX
- endcase

Contoh 1

Diberikan nama dan nilai mahasiswa, jika nilai tersebut lebih besar atau sama dengan 60 maka mahasiswa tersebut dinyatakan lulus jika nilai lebih kecil dari 50 maka dinyatakan tidak lulus.
Bila nilainya 50 sampai dengan 59, maka harus mengikuti remedi.



Bagaimana Pseudo Code ?

- Pseudo code :
 - algoritma kelulusan
 - deklarasi
 - nilai : integer
 - ket : string
 - deklarasi
 - read (nilai)
 - Case nilai of
 - 60..100: ('lulus')
 - 50 .. 59 : ('remidi')
 - 0 .. 49 : ('tidak lulus')
 - endcase
 - Write (keterangan)

Contoh 2:

- Buatlah algoritma dan program yang membaca angka bulan dan tahun, lalu menuliskan jumlah hari dalam bulan tersebut. Misalnya jika dibaca bulan 8 (agustus), maka jumlah harinya adalah 31.

Bulan	Jumlah hari
1,3,5,7,8,10,12	31
4,6,9,11	30
2	29 (jika tahun kabisat), 28 (jika bukan kabisat)

```

Algoritma JUMLAH_HARI
{ menentukan jumlah hari dalam satu bulan }

DEKLARASI
AngkaBulan    : integer      { 1 .. 12 }
Tahun          : integer      { > 0 }
JumlahHari     : integer

DESKRIPSI
read (AngkaBulan,Tahun)
case (AngkaBulan) of
  AngkaBulan= [ 1, 3, 5, 7, 8, 10, 12 ] : JumlahHari=31
  AngkaBulan= [ 4, 6, 9, 11 ]       : JumlahHari=30
  AngkaBulan= 2 : case Tahun
    Tahun mod 4 = 0 : JumlahHari=29
    Tahun mod 4 ≠ 0 : JumlahHari=28
  endcase
endcase

write(JumlahHari)
  
```