

Algoritma dan Struktur Data I

Mia Fitriawati, S.Kom, M.Kom

fppt.com

Pemilihan

- Seringkali suatu instruksi hanya bisa dikerjakan **jika ia memenuhi suatu persyaratan tertentu**
- Komputer tidak lagi mengerjakan instruksi secara **sekuensial**, tetapi berdasarkan **syarat yang dipenuhi**
- Struktur Pemilihan memungkinkan kita **melakukan aksi jika suatu syarat dipenuhi**

fppt.com

Analisis Satu Kasus

- Notasi algoritmik untuk analisis satu kasus menggunakan bentuk berikut:

if *kondisi* then

aksi

endif

- Pernyataan diatas berarti bahwa **aksi** hanya dilaksanakan bila **kondisi** bernilai **benar (true)**
- Bila **kondisi** bernilai **salah (false)** maka **tidak ada aksi** apapun yang dikerjakan

fppt.com

Contoh1: Bilangan Genap

PROGRAM Genap

{ Program untuk mencetak pesan "bilangan genap" jika sebuah bilangan bulat yang dibaca dari piranti masukan merupakan bilangan genap }

DEKLARASI

x : integer

ALGORITMA:

read(x)

if $x \bmod 2 = 0$ then

write ('genap')

endif

fppt.com

Contoh2: Huruf Vokal

PROGRAM Huruf_Vokal

{ Mencetak pesan "huruf vokal" bila sebuah karakter yang dibaca merupakan huruf hidup, asumsikan karakter yang dibaca adalah huruf kecil}

DEKLARASI

c : char

ALGORITMA:

read(c)

if (c='a') or (c='i') or (c='u') or (c='e') or (c='o') then

write ('huruf vokal')

endif

fppt.com

Analisis Dua Kasus

- Notasi algoritmik untuk analisis dua kasus menggunakan bentuk berikut:

if *kondisi* then

aksi1

else

aksi2

endif

- Pernyataan diatas berarti bahwa **aksi1** dilaksanakan bila **kondisi** bernilai **benar (true)**, bila **kondisi** bernilai **salah (false)** maka **aksi2** yang akan dikerjakan

fppt.com

Contoh1: Bilangan Genap/Ganjil

PROGRAM GenapGanjil

{ Mencetak pesan "genap" jika sebuah bilangan bulat yang dibaca merupakan bilangan genap, atau "ganjil" jika bilangan tersebut ganjil }

DEKLARASI

x : integer

ALGORITMA:

```
read(x)
if x mod 2 = 0 then
  write ('genap')
else
  write ('ganjil')
endif
```

fppt.com

Contoh2: Bilangan Terbesar

PROGRAM Maksimum

{ Menentukan bilangan terbesar dari dua buah bilangan bulat }

DEKLARASI

A, B : integer

ALGORITMA:

```
read(A,B)
if A > B then
  write ('Bilangan terbesar = ', A)
else
  write (' Bilangan terbesar = ', B)
endif
```

fppt.com

Contoh3: Upah Karyawan

PROGRAM Upah_Karyawan

{Menentukan upah mingguan seorang karyawan. Masukan yang dibaca dari papan kunci adalah nama karyawan, dan jumlah jam kerja. Keluaran program adalah nama karyawan dan upahnya.}

DEKLARASI

```
nama   : string { nama karyawan }
JJK    : integer { Jumlah Jam Kerja }
lembur : integer { jumlah jam lembur }
upah   : real { nama karyawan }
```

fppt.com

Contoh3: Upah Karyawan (Ljt.)

ALGORITMA:

```
read(nama, JJK)
if JJK ≤ 48 then
    upah ← JJK * 2000
else
    lembur ← JJK - 48
    upah ← 48 * 2000 + lembur * 3000
endif
write (nama, upah)
```

fppt.com

Contoh4: Upah Karyawan 2

PROGRAM Upah_Karyawan

{Menentukan upah mingguan seorang karyawan. Masukan yang dibaca dari papan kunci adalah nama karyawan, dan jumlah jam kerja. Keluaran program adalah nama karyawan dan upahnya.}

DEKLARASI

const JamNormal = 48 { jumlah jam kerja normal/minggu}

const UpahPerJam = 2000 {upah per jam kerja normal}

const UpahLembur = 3000 {upah lembur per jam}

nama : string { nama karyawan }

JJK : integer { Jumlah Jam Kerja }

lembur : real { jumlah jam lembur }

upah : real { upah karyawan }

fppt.com

Contoh4: Upah Karyawan2 (Ljt.)

ALGORITMA:

read(nama, JJK)

if JJK ≤ JamNormal then

 upah ← JJK * UpahPerJam

else

 lembur ← JJK - JamNormal

 upah ← 48 * UpahPerJam + lembur * UpahLembur

endif

write (nama, upah)

fppt.com

Contoh5: Tahun Kabisat

PROGRAM TahunKabisat

{ Menentukan apakah suatu tahun adalah tahun kabisat atau bukan }

DEKLARASI

thn : integer {tahun}

ALGORITMA:

read(thn)

if (thn mod 4 = 0 and thn mod 100≠0) or (thn mod 400 = 0) then

write (' tahun kabisat ')

else

write (' bukan tahun kabisat ')

endif

fppt.com

Analisis Tiga Kasus atau Lebih

- Notasi Algoritmik:

if *kondisi1* then

aksi1

else

if *kondisi2* then

aksi2

else

if *kondisi3* then

aksi3

endif

endif

endif

fppt.com

Contoh1: Jenis Bilangan Bulat

PROGRAM JenisBilanganBulat

{ Menentukan jenis bilangan bulat: positif, negatif, atau nol }

DEKLARASI

x : integer

ALGORITMA:

read(x)

if x > 0 then

write (' positif ')

else

if x < 0 then

write (' negatif ')

else

write (' nol ')

endif

endif

fppt.com

Contoh2 : Upah Per Golongan

- Algoritma menghitung upah karyawan berdasarkan golongannya:
 - Golongan A, upah per jam Rp. 4000,-
 - Golongan B, upah per jam Rp. 5000,-
 - Golongan C, upah per jam Rp. 6000,-
 - Golongan D, upah per jam Rp. 7000,-
- Kelebihan jam kerja dihitung lembur dengan upah lembur per jam sama untuk semua golongan
- Algoritma membaca nama karyawan dan jumlah jam kerjanya selama seminggu dan menghitung gaji mingguannya

fppt.com

Jawaban: Upah Per Golongan

PROGRAM UpahKaryawanGolongan

{ Menentukan upah mingguan seorang karyawan. Masukan yang dibaca dari papan kunci adalah nama karyawan, golongan, dan jumlah jam kerja. Keluaran program adalah nama karyawan dan upahnya.}

DEKLARASI

const JamNormal = 48 { jumlah jam kerja normal/minggu}

const UpahLembur = 3000 {upah lembur per jam}

Nama : string { nama karyawan }
 gol : char { gol karyawan: 'A', 'B', 'C', atau 'D' }
 JJK : integer { Jumlah Jam Kerja }
 JamLembur : integer { jumlah jam lembur }
 UpahPerJam : real { upah per jam }
 UpahTotal : real

fppt.com

Jawaban: Upah Per Gol. (Ijt.)

ALGORITMA:

```

read (Nama, gol, JJK)
if gol = 'A' then
  UpahPerJam ← 4000.0
else
  if gol = 'B' then
    UpahPerJam ← 5000.0
  else
    if gol = 'C' then
      UpahPerJam ← 6000.0
    else
      if gol = 'D' then
        UpahPerJam ← 7000.0
      endif
    endif
  endif
endif
endif

```

fppt.com

Jawaban: Upah Per Gol. (ljt.)

ALGORITMA:

```

.....
if JJK ≤ JamNormal then
    UpahTotal ← JJK * UpahPerJam
else
    JamLembur ← JJK – JamNormal
    UpahTotal ← JamNormal*UpahPerJam +
JamLembur*UpahLembur
endif
write (Nama, UpahTotal)

```

fppt.com

Struktur CASE

- Untuk menyederhanakan **analisis kasus** untuk **masalah dengan dua kasus atau lebih**, dapat digunakan struktur CASE berikut ini:

```

case ekspresi
    nilai1 : aksi1
    nilai2 : aksi2
    nilai3 : aksi3
    ....
    nilai_n : aksi_n
    otherwise : aksi_x
endcase

```

fppt.com

Contoh1: Bilangan Genap/Ganjil

PROGRAM GenapGanjil

{ Mencetak pesan "genap" jika sebuah bilangan bulat yang dibaca merupakan bilangan genap, atau "ganjil" jika bilangan tersebut ganjil}

DEKLARASI

x : integer

ALGORITMA:

read(x)

case x mod 2 = 0

0 : write ('genap')

1 : write ('ganjil')

endcase

fppt.com

Contoh2 : Upah Per Golongan

PROGRAM UpahKaryawanGolongan

{ Menentukan upah mingguan seorang karyawan. Masukan yang dibaca dari papan kunci adalah nama karyawan, golongan, dan jumlah jam kerja. Keluaran program adalah nama karyawan dan upahnya.}

DEKLARASI

const JamNormal = 48 { jumlah jam kerja normal/minggu}

const UpahLembur = 3000 {upah lembur per jam}

Nama : string { nama karyawan }

gol : char { gol karyawan: 'A', 'B', 'C', atau 'D' }

JJK : integer { Jumlah Jam Kerja }

JamLembur : integer { jumlah jam lembur }

UpahPerJam : real { upah per jam }

UpahTotal : real

fppt.com

Contoh2: Upah Per Gol. (ljt.)

ALGORITMA:

```

read (Nama, gol, JJK)
case (gol)
  'A' : UpahPerJam ← 4000.0
  'B' : UpahPerJam ← 5000.0
  'C' : UpahPerJam ← 6000.0
  'D' : UpahPerJam ← 7000.0
endcase

```

fppt.com

Contoh2: Upah Per Gol. (ljt.)

ALGORITMA:

```

.....
if JJK ≤ JamNormal then
  UpahTotal ← JJK * UpahPerJam
else
  JamLembur ← JJK – JamNormal
  UpahTotal ← JamNormal*UpahPerJam +
  JamLembur*UpahLembur
endif
read (Nama, UpahTotal)

```

fppt.com

Contoh3 : Pemilihan Menu

PROGRAM EmpatPersegiPanjang

{ Menampilkan menu perhitungan empat persegi panjang, memilih menu dan melakukan proses perhitungan.}

DEKLARASI

NomorMenu : integer

panjang, lebar : real

luas, keliling, diagonal : real

fppt.com

Contoh3: Pemilihan Menu (ljt.)

ALGORITMA:

{cetak menu}

write (' MENU EMPAT PERSEGI PANJANG ')

write (' 1. Hitung luas ')

write (' 2. Hitung keliling ')

write (' 3. Hitung panjang diagonal ')

write (' 4. Keluar program ')

write (' Masukkan pilihan anda (1/2/3/4)? ')

read (NomorMenu)

fppt.com

Contoh3: Pemilihan Menu (ljt.)

ALGORITMA:

.....

```

case NomorMenu
  1 : read(panjang,lebar)
      luas ← panjang * lebar
      write (luas)
  2 : read(panjang,lebar)
      keliling ← 2*panjang + 2*lebar
      write (keliling)
  3 : read(panjang,lebar)
      diagonal ← sqrt(panjang*panjang + lebar*lebar)
      write (diagonal)
  4 : write ('Keluar program....sampai jumpa')
endcase

```

fppt.com

Contoh4 : Jumlah hari

PROGRAM JumlahHari

{ Menentukan jumlah hari dalam satu bulan.}

DEKLARASI

```

nomor_bulan : integer {1 . . 12}
tahun : integer { > 0}
jumlah_hari : integer

```

fppt.com

Contoh4: Jumlah hari (ljt.)

ALGORITMA:

```

read(nomor_bulan, tahun)
case nomor_bulan
  1, 3, 5, 7, 8, 10, 12 : jumlah_hari ← 31
  4, 6, 9, 11          : jumlah_hari ← 30
  2 : if (tahun mod 4 = 0 and tahun mod 100≠0) or
      (tahun mod 400 = 0) then
      jumlah_hari ← 29
      else
      jumlah_hari ← 28
      endif
endcase
write (jumlah_hari)

```

fppt.com

Latihan

1. Buatlah sebuah algoritma mencetak pesan "bilangan genap" kemudian kalikan bilangan tersebut dengan angka 2 jika bilangan bulat yang dimasukkan dari piranti masukan merupakan bilangan genap.
2. Buatlah sebuah algoritma mengenai nilai kelulusan mahasiswa. Menerima masukkan nama dan nilai mahasiswa, jika nilai tersebut sama dengan 60 maka mahasiswa tersebut dinyatakan lulus. Jika tidak maka dinyatakan tidak lulus.
3. Buatlah sebuah algoritma untuk membaca dua buah nilai integer, jika nilai pertama lebih besar atau sama dengan nilai kedua kerjakan nilai pertama/nilai kedua, jika nilai kedua lebih besar dari nilai pertama kerjakan nilai kedua/nilai pertama, tetapi jika nilai kedua = 0 maka tampilkan pesan error
4. Buatlah algoritma untuk membaca nomor 1-7 untuk menunjukkan hari. Tuliskan nama hari sesuai dengan nomor harinya.

fppt.com

