



# Skema Standar (Bag. 1)

## LATIHAN SOAL

Tim Pengajar

IF1210 Dasar Pemrograman

Sem. 2 2019/2020



# Petunjuk Pengerjaan

- Latihan soal ini dikerjakan secara individual. Tuliskan pada bagian header: NIM, nama, kelas
- Anda diperkenankan berdiskusi dengan rekan lain, tapi harap mengerjakan sendiri-sendiri. ***Dilarang copy-paste pekerjaan orang lain!*** Tuliskan nama dan NIM rekan diskusi Anda di lembar jawaban Anda.
- Jawaban disubmit dalam bentuk file pdf melalui link yang disediakan di situs Olympia mulai hari Kamis 19 Maret 2020 jam 09.00 s.d. jam 23.59.
- Setiap mahasiswa hanya perlu men-submit 1 file saja.



# Latihan 1. Skema Proses Validasi

- Buatlah program dalam notasi algoritmik yang menerima 3 buah bilangan integer yaitu h, m, dan s yang akan digunakan untuk membentuk data bertipe jam. Definisi type jam adalah sbb.

```
type jam : < HH : integer[0..23]; { bagian jam }  
           MM : integer[0..59]; { bagian menit }  
           SS : integer[0..59] > { bagian detik }
```

- Jika ketiga input valid, maka sebuah variabel J bertipe jam akan terbentuk (didefinisikan nilainya) dengan J.HH bernilai h, J.MM bernilai m, J.SS bernilai s.
- Nilai valid didefinisikan sebagai:  $0 \leq h \leq 23$ ;  $0 \leq m \leq 59$ ;  $0 \leq s \leq 59$
- Jika tidak valid, dituliskan pesan kesalahan ke layar “Tidak dapat membentuk jam”

# Latihan 2.



## Skema Pemrosesan Sekuensial (1)

- Buatlah program dalam notasi algoritmik, untuk membaca sejumlah nilai UTS mahasiswa di suatu kelas. Nilai UTS yang valid adalah 0..100. Pembacaan dihentikan jika **masukan nilai UTS di luar range nilai yang diizinkan**
- Di akhir program dihitung dan ditampilkan rata-rata nilai UTS seluruh mahasiswa di kelas.
- Jika tidak ada nilai UTS yang dimasukkan, tuliskan “Tidak ada data”
- Contoh... (lihat slide berikutnya)



# Latihan 2.

## Skema Pemrosesan Sekuensial (2)

Contoh masukan dan keluaran:

Catatan: yang bergaris bawah adalah masukan pengguna

### Contoh 1

Nilai UTS = 50

Nilai UTS = 100

Nilai UTS = 9999

Nilai rata-rata UTS = 75

### Contoh 2

Nilai UTS = 101

Tidak ada data

Diskusi:  
Skema apa yang  
sebaiknya digunakan

# Latihan 3.

## Skema Proses Validasi

- Buatlah program dalam notasi algoritmik yang menerima 3 buah bilangan integer yaitu h, m, dan s yang akan digunakan untuk membentuk data bertipe jam. Definisi type jam adalah sbb.

```
type jam : < HH : integer[0..23]; { bagian jam }  
           MM : integer[0..59]; { bagian menit }  
           SS : integer[0..59] > { bagian detik }
```

- Jika ketiga input tidak valid**, dituliskan pesan kesalahan ke layar “Tidak dapat membentuk jam” dan pemasukan data h, m, s diulangi sampai didapatkan nilai yang valid.
- Jika ketiga input valid**, maka sebuah variabel J bertipe jam akan terbentuk (didefinisikan nilainya) dengan J.HH bernilai h, J.MM bernilai m, J.SS bernilai s.
- Nilai valid didefinisikan sebagai:  $0 \leq h \leq 23$ ;  $0 \leq m \leq 59$ ;  $0 \leq s \leq 59$

# Latihan 4



- Buatlah program dalam notasi algoritmik yang menerima tiga buah nilai resistor R1, R2, dan R3, berupa bilangan integer  $\geq 0$ , dan menghitung nilai resistansi total. Ketiga resistor tersebut dapat dihubungkan secara serial maupun paralel.
- Pengguna bisa memilih untuk menghitung resistansi total (RT):
  - Jika dihubungkan serial (pilihan 1), yaitu  $RT = R1 + R2 + R3$ ; dan
  - Jika dihubungkan paralel (pilihan 2), yaitu  $1/RT = 1/R1 + 1/R2 + 1/R3$ .
- Buatlah program yang memvalidasi semua masukan sehingga didapatkan masukan yang tepat dan selanjutnya menghitung dan menampilkan resistansi total.
- Validasi input pilihan pengguna, dan validasi input nilai resistor dari pengguna jika nilai resistor dihubungkan secara paralel (yaitu nilai resistor tidak boleh  $\leq 0$ )



# Latihan 5 (1)

- Buatlah program dalam notasi algoritmik, untuk membaca nilai UTS dan nilai UAS mahasiswa untuk setiap pelajaran yang diikutinya (0..100) **dan diakhiri jika nilai masukan UTS di luar range nilai yang diizinkan**, kemudian menghitung dan mencetak rata-rata nilai akhir dari seluruh pelajaran.
- Gunakan skema validasi data untuk memastikan nilai UAS pada range 0..100 (jika data tidak memenuhi syarat, read data UAS diulang). Nilai akhir untuk suatu pelajaran dihitung dari rumus  $(40\% * \text{nilai UTS}) + (60\% * \text{nilai UAS})$ .
- Contoh masukan dan keluaran (slide berikutnya):





# Latihan 5 (2)

Contoh masukan dan keluaran:

Catatan: yang bergaris bawah adalah input pengguna

## Contoh 1

Nilai UTS = 50

Nilai UAS = 200

Ulangi input nilai (0..100)!

Nilai UAS = 100

Nilai akhir pelajaran 1 = 80

Nilai UTS = 100

Nilai UAS = 50

Nilai Akhir pelajaran 2 = 70

Nilai UTS = 9999

Nilai rata-rata dari 2 pelajaran adalah = 75

## Contoh 2

Nilai UTS = 101

Data kosong, tidak ada nilai rata-rata !



# Latihan 6

- Buatlah program dalam notasi algoritmik yang membaca P1 dan P2 bertipe pecahan. Berikut definisi type pecahan di notasi algoritmik:

```
type pecahan : < pembilang : integer;  
                penyebut : integer > 0 >
```

- Program akan membandingkan nilai pecahan P1 dan P2 dan menghitung selisihnya (dalam tipe **pecahan**), kemudian menuliskan hasilnya di layar.
- Sebelum melakukan proses perbandingan, program harus memeriksa bahwa pengguna memasukkan nilai pecahan yang valid, yaitu penyebut bernilai lebih besar daripada nol.
- Jika masukan tidak valid, program menampilkan pesan kesalahan di layar “Masukan tidak valid”. Masukan tidak perlu diulangi.