

JAM (1)

Judul

{ Membaca bilangan dari keyboard, bilangan ditampung dalam variabel h, m, s.
Nilai dari variabel h memiliki range [0..23], variabel m dan s memiliki range [0..59], Nilai variabel h, m dalam struktur komposisi Jam yang terdiri dari < HH : MM : SS >.
Diketahui Variabel J bertipe Jam, maka J.HH, J.MM, J.SS dapat diakses }

Kamus

Type
Jam = Record
 HH, MM, SS : Integer
end

J : Jam
h,m,s : integer

ALGORITMA

```
input (h)  { Valid jika 24 > bilangan >=0 }
input (m)  { Valid jika 60 > bilangan >=0 }
input (s)  { Valid jika 60 > bilangan >=0 }
J.HH ← h   J.MM ← m   J.SS ← s
output("Nilai Jam : ", J )
```

aetthea, teknik elektro unsoed

JAM (2)

Judul

{ Membaca bilangan dari keyboard, bilangan ditampung dalam variabel h, m, s.
Nilai dari variabel h memiliki range [0..23], variabel m dan s memiliki range [0..59],
Nilai variabel h, m, s akan dikonstruksikan dalam struktur komposisi Jam yang
terdiri dari < HH : MM : SS >.
Diketahui Variabel J bertipe array Jam, maka J_i.HH, J_i.MM, J_i.SS dapat diakses }

Kamus

Type
Jam = Record
 HH, MM, SS : Integer
end
h,m,s,i : integer
J : Array [1.. N] of Jam
Jml : Integer { Jumlah Elemen Array yang akan diisi, range jml adalah [1.. N] }

ALGORITMA

```
input(jml) { Valid jika 0 < bilangan <=N }
while ( i <= jml) do
    input (h)  { Valid jika 24> bilangan >=0 }
    input (m)  { Valid jika 60> bilangan >=0 }
    input (s)  { Valid jika 60> bilangan >=0 }
    Ji.HH ← h   Ji.MM ← m   Ji.SS ← s
    output("Nilai Jam : ", Ji )
    i ← i+1
```

aetthea, teknik elektro unsoed

JAM (3)

Judul

{ Membaca bilangan dari keyboard, bilangan ditampilkan dalam variabel h, m, s.
Nilai dari variabel h memiliki range [0..23], variabel m dan s memiliki range [0..59], Nilai variabel h, m, s akan dikonstruksikan dalam struktur komposisi Jam yang terdiri dari < HH : MM : SS >.
Diketahui Variabel J bertipe Jam, maka J.HH, J.MM, J.SS dapat diakses

Operasi lain yang dapat dilakukan terhadap Jam adalah operasi :

1. Membentuk sebuah nilai bertipe Jam dengan <0 : 0 : 0>
2. Tambah jam, menambahkan jumlah menit ke suatu jam current (saat ini).
Misal jam <2:5:3>
tambah 30 menit maka hasilnya <2:35:3>, jam <2:5:3> tambah 75 menit maka hasilnya <3:20:3>
3. Konversikan jam ke detik, misal jam <1:4:5> menjadi 3845 detik
4. Konversikan detik ke jam, misalnya 4185 detik menjadi jam <1:9:45>
5. Membaca Jam, menerima masukkan nilai jam dari pengguna program
6. Menuliskan Jam, menulis nilai jam dengan format HH:MM:SS
7. Next Jam, memajukan komponen HH plus 1
8. Next Minute, memajukan komponen MM plus 1
9. Next Detik, memajukan komponen SS plus 1 }

Kamus

Type

Jam = Record
HH, MM, SS : Integer
end

J : Jam
h,m,s : integer }

aetthea, teknik elektro unsoed

JAM (3)

ALGORITMA

```
{ Create Jam }
.....
{ Baca Jam }
.....
{ Tulis Jam }
.....
{ Tambah Jam }
.....
{ Konversi Jam to Detik }
.....
{ Konversi detik to jam }
.....
{ Next jam }
.....
{ Next Menit }
.....
{ Next Detik }
```

Tugas : Mahasiswa berkelompok menjadi 9 kelompok, setiap kelompok membuat algoritma setiap operasi yang dikehendaki, setelah selesai harap dituliskan di papan tulis.

Catatan : Pengalaman belajar tersebut akan diarahkan untuk memahami topik Procedure dan Function

aetthea, teknik elektro unsoed

JAM (4)

-----Procedure dan Function-----

```
{ Create Jam }
Procedure CreateJam()
{ Men-set nilai jam menjadi <0:0:0>
Kamus :
Algoritma :
    J.HH ← 0   J.MM ← 0   J.SS ← 0

{ Baca Jam }
Procedure BacaJam ( out JJ : Jam )
{ Membaca nilai h,m,s dari pengguna, mengkonstruksikannya dalam
  nilai Jam }
Kamus Lokal :
    h,m,s : integer
Algoritma :
    input (h, m, s)
    JJ.HH ← h   JJ.MM ← m   JJ.SS ← s

{ Tulis Jam }
.....
{ Tambah Jam }
.....
{ Konversi Jam to Detik }
.....
{ Konversi detik to jam }      aetthea, teknik elektro unsoed
.....
{ Next jam }
```

JAM (4)

-----Procedure dan Function-----

```
{ Tulis Jam }
Procedure TulisJam(in JJ : Jam)
{ Menuliskan komponen jam ke layar dalam format <HH:MM:SS>
Kamus :
Algoritma :
    output("<", JJ.HH, ":", JJ.MM, ":", JJ.SS, ">")

{ Konversi Jam To Detik }
Function JamToDetik(in JJ : Jam) → integer
{ Mengkonversikan nilai Jam ke detik dengan Formula sbb :
  → (JJ.HH * 3600 + JJ.MM * 60 + JJ.SS) }
Kamus :
Algoritma :
    → (JJ.HH * 3600 + JJ.MM * 60 + JJ.SS)

.....
{ Tambah Jam }
.....
{ Konversi Jam to Detik }
.....
{ Konversi detik to jam }
.....
{ Next jam }      aetthea, teknik elektro unsoed
.....
{ Next Menit }
```

JAM (4)

-----Procedure dan Function-----

```
{ Konversi Detik To Jam }
Function DetikToJam(in SS : integer) → Jam
{ Mengkonversikan nilai detik ke Jam dengan Formula sbb :
  h ← (SS / 3600)
  Sisa ← (SS % 3600)
  m ← Sisa / 60
  s ← Sisa % 60
}
Kamus :
  h, m, s, Sisa : Integer
  JJ           : Jam
Algoritma :
  h ← (SS / 3600)
  Sisa ← (SS % 3600)
  m ← Sisa / 60
  s ← Sisa % 60
  JJ.HH ← h  JJ.MM ← m  JJ.SS ← s
  → JJ
```

```
.....
{ Tambah Jam }          aetthea, teknik elektro unsoed
.....
{ Konversi Jam to Detik }
```

JAM (4)

-----Procedure dan Function-----

```
{ Tambah Jam }
Procedure TambahJam( in out JJ : Jam, in m : integer)
{ Menambahkan JJ dengan m dengan langkah berikut :
  JJdetik ← JJ dikonversikan ke detik
  JJm     ← m dikonversikan ke detik
  Hasil ← JJdetik + JJm
  JJ ← Hasil dikonversikan ke Jam
}
Kamus :
  JJdetik, JJm, Hasil : Integer
Algoritma :
  JJdetik ← JamToDetik(JJ)
  JJm     ← m * 60
  Hasil ← JJdetik + JJm
  JJ ← DetikToJam(Hasil)
```

```
.....
{ Tambah Jam }          aetthea, teknik elektro unsoed
.....
{ Konversi Jam to Detik }
```