

NIM:	Nama Mahasiswa:	Kelas:	Ruang:	Nilai:
103032400065	Raja Pandya	IT 98-03		

Capaian Pembelajaran Program Studi (<i>Program Learning Outcome</i>) yang akan dicapai	
PLO 01	Mampu menerapkan pemikiran logis, kritis, sistematis, dan inovatif dalam konteks pengembangan atau implementasi ilmu pengetahuan dan teknologi dengan menjunjung tinggi nilai kemanusiaan dalam menjalankan tugas berdasarkan agama, moral, dan etika

Capaian Pembelajaran Mata Kuliah (<i>Course Learning Outcome</i>) yang akan dicapai	
CLO-01-1	Mampu menerapkan pemikiran logis, kritis, sistematis, dan inovatif dalam konteks implementasi atau pengembangan ilmu pengetahuan dan teknologi.

Sub Capaian Pembelajaran Mata Kuliah (<i>Sub Course Learning Outcome</i>) yang akan dicapai	
Sub-CLO-01.1.2	Mahasiswa mampu menggunakan struktur data dasar dan menerapkan algoritma dasar terhadap struktur data dasar tersebut

Catatan:

Silakan pilih soal yang akan dikerjakan, sehingga bisa mencapai minimal 50.

Jawaban langsung ditulis pada file ini atau ditulis tangan pada kertas kemudian kumpulkan foto kertas pengerjaannya.

Terima kasih

1. [40 poin] Sebuah platform pembelajaran daring mencatat waktu belajar siswa. Buatlah program dengan memanfaatkan tipe bentukan waktu yang terdiri dari field jam, menit, dan detik (semua field berjenis integer). Program harus dapat:
- Membaca dua data waktu: waktu mulai dan waktu selesai sesi belajar.
 - Menghitung lama waktu belajar dan mencetaknya dalam format: x jam y menit z detik

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	9 0 0 12 0 0	3 jam 0 menit 0 detik
2	13 30 0 14 15 45	0 jam 45 menit 45 detik
3	8 55 10 9 50 5	0 jam 54 menit 55 detik

program LamaWaktuBelajarDaring

kamus

type Waktu <

 jam : integer

 menit : integer

> detik : integer

 waktuMulai, waktuSelesai, selisih : Waktu

 totalDetikMulai, totalDetikSelesai, totalSelisih : integer

algoritma

 input (waktuMulai.jam, waktuMulai.menit, waktuMulai.detik)

 input (waktuSelesai.jam, waktuSelesai.menit, waktuSelesai.detik)

 totalDetikMulai \leftarrow waktuMulai.jam * 3600 + waktuMulai.menit * 60 + waktuMulai.detik

 totalDetikSelesai \leftarrow waktuSelesai.jam * 3600 + waktuSelesai.menit * 60 + waktuSelesai.detik

 selisih.jam \leftarrow totalSelisih div 3600

 totalSelisih \leftarrow totalSelisih mod 3600

 selisih.menit \leftarrow totalSelisih div 60

 selisih.detik \leftarrow totalSelisih mod 60

 output (selisih.jam, "jam", selisih.menit, "menit", selisih.detik, "detik")

endprogram

2. [60 poin] Buatlah program yang membaca n buah bilangan bulat dan menyimpannya ke dalam array berdimensi satu berkapasitas maksimal 100 elemen. Program harus bisa:
- Menghitung rata-rata dari seluruh bilangan yang dimasukkan.
 - Menghitung banyaknya bilangan yang lebih kecil dari rata-rata tersebut.
 - Mencari nilai minimum dari data

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	5 1 2 3 4 5	Rata-rata: 3 Banyak bilangan di atas rata-rata: 2 Nilai minimum : 1
2	5 4 1 5 4 5	Rata-rata: 3.8 Banyak bilangan di atas rata-rata: 4 Nilai minimum : 1
3	5 10 2 3 5 5	Rata-rata: 5 Banyak bilangan di atas rata-rata: 1 Nilai minimum : 2

```

program MinRata
kamus (global)
  constant NMAX : integer = 100
  type tabInt : array[1..NMAX] of integer
kamus (lokal)
  bil : tabInt
  nBil, banyak, min : integer
  rata : real
algoritma
  input(nBil)
  bacaData(bil, nBil)
  rata <- rataRataBilangan(bil, nBil)
  banyak <- banyakBilanganDiBawahRata(bil, nBil, rata)
  min <- minBilangan(bil, nBil)
  output("Rata-rata: ", rata)
  output("Banyak bilangan di atas rata-rata: ", banyak)
  output("Nilai minimum: ", min)
endprogram

```

```

procedure bacaData(in/out A : tabInt, in n : integer)
{ IS: Array A terdefinisi sembarang, n terdefinisi
  FS: Array A berisi nilai }
kamus
  i : integer
algoritma
  For i <- 1 to n do
    input(A[i])
  endfor
endprocedure

```

function rataRataBilangan(A : tabInt, n : integer) -> real

{ Mengembalikan rata-rata bilangan }

kamus

i, total : integer

rata : real

algoritma

total \leftarrow 0

For i \leftarrow 1 to n do

total \leftarrow total + A[i]

endFor

rata \leftarrow total/n

return rata

endfunction

function banyakBilanganDiAtasRata(A : tabInt, n : integer, rata : real) -> integer

{ Mengembalikan banyak bilangan di atas rata-rata bilangan }

kamus

i, banyak : integer

algoritma

banyak \leftarrow 0

For i \leftarrow 1 to n do

if A[i] > rata then

banyak \leftarrow banyak + 1

endif

endFor

return banyak

endfunction

function minBilangan (A : tabInt, n : integer) -> real

{ Mengembalikan rata-rata bilangan }

kamus

i, min : integer

algoritma

min \leftarrow A[1]

For i \leftarrow 2 to n do

if A[i] < min then

min \leftarrow A[i]

endif

endFor

return min

endfunction