



# **MODUL PRAKTIKUM PEMROGRAMAN KOMPUTER**

**FISIKA  
UNIVERSITAS NEGERI JAKARTA**

# MODUL 1

## SKALA RICHTER



### Praktikum Pemrograman Komputer 115

Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Implementasi Storing and Values
2. Menerapkan Decision Making

#### B. PERSOALAN

Skala Richter digunakan untuk mengklasifikasi descriptor dari gempa bumi. Berikut adalah tabel yang menunjukkan skala Richter dengan descriptornya.

Magnitude	Descriptor
Less than 2.0	Micro
2.0 to less than 3.0	Very Minor
3.0 to less than 4.0	Minor
4.0 to less than 5.0	Light
5.0 to less than 6.0	Moderate
6.0 to less than 7.0	Strong
7.0 to less than 8.0	Major
8.0 to less than 10.0	Great
10.0 or more	Meteoric

Buat pemrograman untuk membaca input nilai skala Richter dari user. Program akan menampilkan klasifikasi descriptor tersebut. Buat pilihan di akhir pembacaan, seperti pada contoh output.

## C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat 5 langkah pemrograman

1. Problem Statement
2. Mathematical Equation (khusus untuk Modul 1, mathematical equation dapat dituliskan condition)
3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)
4. Flowchart
5. Source Code

Contoh Output

Modul 1 Klasifikasi Skala Richter

Nama : xxxxxxxxxxxxxxxx

NRM : 99999999999999

Masukan besar skala Richter = **3.7**

Deskriptor gempa : Minor

Akhiri (Y/N)? **Y**

Masukan besar skala Richter = **7.7**

Deskriptor gempa : Major

Akhiri (Y/N)? **N**

Terimakasih, semoga aman selalu.

# MODUL 2

## GROSS PAY TABLE



### Praktikum Pemrograman Komputer 115

Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Print Format
2. While Loop
3. Increment

#### B. PERSOALAN

Gross pay merupakan istilah “gaji kotor” yaitu jumlah uang yang diperoleh karyawan dalam periode waktu tertentu sebelum dipotong. Ada banyak cara berbeda untuk menghitung gaji kotor tergantung pada bagaimana seorang karyawan dibayar. Perhatikan data tabel berikut.

Jumlah Jam Kerja (h)	Honor per Jam
$h \leq 40$ jam	Rp 97000,00/ jam
$40 < h \leq 80$	40 jam pertama: Rp 97000,00/ jam  Setiap jam berikutnya s.d. jam ke-80: Rp 103000,00/ jam
$80 < h \leq 100$	40 jam pertama: Rp 97000,00/ jam  Setiap jam berikutnya s.d. jam ke-100: Rp 107500,00/jam
$h > 100$	Yang dibayarkan hanya 100 jam pertama.

Buat pemrograman untuk menampilkan tabel grosspay dengan kelipatan per 5 jam untuk rentang jam kerja 5 s.d. 110 jam.

### C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat 5 langkah pemrograman

1. Problem Statement
2. Mathematical Equation
3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)
4. Flowchart
5. Source Code

#### Contoh Output

Modul 2 Gross Pay

Nama : xxxxxxxxxxxxxxxxx

NRM : 99999999999999

#### TABEL GROSSPAY

No.	Jumlah jam	Gross Pay
1	5	Rp 485.000,00
2	10	.
3	15	.
.	.	.
.	.	.
22	110	Rp 10.330.000,00

# MODUL 3

## COSINUS FUNCTION



### Praktikum Pemrograman Komputer 115

Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Print Format
2. While Loop
3. Increment
4. Def Function

#### B. PERSOALAN

Pada modul ini, ada dua fungsi cosinus yang dibuat atau digunakan. Pertama, fungsi cosinus yang dibangun dari deret Maclaurin, dan yang kedua, fungsi cosinus bawaan dari NumPy package.

Deret Maclaurin untuk fungsi cosinus:

$$\begin{aligned}\cos x &= f(0)\frac{x^0}{0!} + f'(0)\frac{x^1}{1!} + f''(0)\frac{x^2}{2!} + f'''(0)\frac{x^3}{3!} + f^{(4)}(0)\frac{x^4}{4!} + f^{(5)}(0)\frac{x^5}{5!} + f^{(6)}(0)\frac{x^6}{6!} + \dots \\ &= \frac{x^0}{0!} - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots \\ &= \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}\end{aligned}$$

Program berhenti jika  $Er$  (Relative Error)  $< 5\%$

$$Er = \left| \frac{AV - TV}{TV} \right| \times 100\%$$

TV (True Value sin/cos) diperoleh dari perhitungan menggunakan fungsi dari Numpy Package.

AV (Approximation Value) diperoleh dengan menghitung nilai cos menggunakan pendekatan deret Maclaurin.

Input dari user adalah sudut dalam derajat. Untuk dapat diterapkan ke dalam dua fungsi cosinus tersebut, sudut harus dikonversi terlebih dulu menggunakan sebuah fungsi yang terdefinisi. Konversi ini mengubah sudut dalam derajat ke dalam radian.

$$\text{Radians} = \left( \frac{\pi}{180^\circ} \right) \times \text{degrees}$$

$$\text{Degrees} = \left( \frac{180^\circ}{\pi} \right) \times \text{radians}$$

Gunakan nilai pi yang terdapat pada NumPy package untuk mengonversi ini.

**Hint:** Gunakan dua fungsi terdefinisi, 1) fungsi konversi sudut, 2) fungsi cosinus dari deret Maclaurin.

### C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat 5 langkah pemrograman

1. Problem Statement
2. Mathematical Equation
3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)
4. Flowchart
5. Source Code

#### Contoh Output

```
Program Cosinus
Nama : xxxxxxxxxxxxxxxxxxxxxx
NRM  : 9999999999999999
```

Besar sudut (derajat) = ?

True value cos(.....) = .....

Jumlah Suku	AV cosinus	ER cosinus
1	.....	.....%
2	.....	.....%
...	.....	.....%

Mau menghitung lagi (y/t) ?

# MODUL 4

## DETERMINAN MATRIKS



### Praktikum Pemrograman Komputer 115

Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Loop
2. Print Format
3. List from Default Python
4. Array from Numpy Library
5. Def Function

#### B. PERSOALAN

Perhatikan langkah penentuan determinan matriks 3\*3 berikut.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\begin{aligned} |A| &= \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11} \begin{vmatrix} \square & \square & \square \\ \square & a_{22} & a_{23} \\ \square & a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} \square & \square & \square \\ a_{21} & \square & a_{23} \\ a_{31} & \square & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} \square & \square & \square \\ a_{21} & a_{22} & \square \\ a_{31} & a_{32} & \square \end{vmatrix} \\ &= a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \end{aligned}$$

$$|A| = a_{11} \cdot (a_{22} \cdot a_{33} - a_{32} \cdot a_{23}) - a_{12} \cdot (a_{21} \cdot a_{33} - a_{31} \cdot a_{23}) + a_{13} \cdot (a_{21} \cdot a_{32} - a_{31} \cdot a_{22})$$

Pada Modul 4 ini, silakan membuat program menentukan determinan matriks 3\*3 tersebut dengan 2 teknis berbeda, menggunakan List bawaan default Python dan menggunakan Array serta library Numpy.

## C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat 5 langkah pemrograman

1. Problem Statement
2. Mathematical Equation
3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)
4. Flowchart
5. Source Code

### Contoh Output

```
Program Determinan Matriks 3*3
Komparasi Menggunakan List dan Numpy
Nama : xxxxxxxxxxxxxxxxxxxxxx
NRM  : 9999999999999999
```

```
Determinan matriks A menggunakan List adalah 18
Determinan matriks A menggunakan Numpy adalah 18.0
```

```
Mau menghitung lagi (y/t) ?
```

# MODUL 5

## GRADING



### Praktikum Pemrograman Komputer 115

Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Loop
2. Print Format
3. Counting

#### B. PERSOALAN

Membuat program yang menghitung nilai akhir dan grading dari sekumpulan data siswa.

Berikut contoh data siswa, yang terdiri dari nama, nilai quiz, nilai uts, dan nilai uas:

```
Da_Sis = [['Adi', 80,80,86],  
          ['Budi', 65,80,70],  
          ['Cepi', 60,60,70],  
          ['Dedi', 50,70,50],  
          ['Abu', 90,80,70],  
          ['Baskara', 65,80,75],  
          ['Elang', 55,60,55]]
```

Pada Modul 5 ini, silakan membuat program untuk menghitung nilai akhir dari data siswa dan menentukan grade-nya.

Perhitungan nilai akhir mengikuti rumus berikut:

$$NA = 30\% \text{ Quiz} + 35\% \text{ UTS} + 35\% \text{ UAS}$$

Adapun ketentuan grading mengikuti tabel berikut.

Tingkat Penguasaan (%)	Huruf	Angka	Keterangan
> 85 - 100	A	4	Lulus
> 80 - 85	A-	3.7	Lulus
> 75 - 80	B+	3.3	Lulus
> 70 - 75	B	3.0	Lulus
> 65 - 70	B-	2.7	Lulus
> 60 - 65	C+	2.3	Lulus
> 55 - 60	C	2.0	Lulus
> 50 - 55	C-	1.7	Belum Lulus
> 45 - 50	D	1	Belum Lulus
0 - 45	E	0	Belum Lulus

### C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat 5 langkah pemrograman

1. Problem Statement
2. Mathematical Equation
3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)
4. Flowchart
5. Source Code
6. Bukti Running untuk beberapa data (silakan dibuat minimal 3 variasi, agar mewakili semua rentang grade)

Jika program dijalankan dengan contoh data, maka luarannya adalah sebagai berikut.

Program Grading

Nama : xxxxxxxxxxxxxxxxxxxxxx

NRM : 9999999999999999

LAPORAN GRADING MAHASISWA

```
-----  
No | Nama          | Nilai   | Grade  |  
   | Mahasiswa    | Akhir  |        |  
---|-----|-----|-----|  
1  | Adi           | 82     | A-     |  
2  | Budi          | 72     | B      |  
3  | Cepi          | 63     | C+     |  
4  | Dedi          | 57     | C      |  
5  | Abu           | 80     | A-     |  
6  | Baskara      | 73     | B      |  
7  | Elang         | 57     | C      |  
-----
```

Rekapitulasi :

Grade A = 2

Grade B = 2

Grade C = 1

Grade D = 2

Total Mahasiswa = 7

\*selesai\*

# MODUL 6

## SORTING



### Praktikum Pemrograman Komputer 115

Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Decision
2. Loop
3. Kombinasi decision dan loop untuk membangun algoritma sorting

#### B. PERSOALAN

Algoritma sorting terdiri dari beberapa jenis. Dalam modul ini, mahasiswa diminta membuat 2 algoritma (termasuk flowchart dan code) untuk:

1. Buble Sort
2. Selection Sort

Silakan gunakan data berikut.

```
Nilai = [85, 13, 99, 34, 71, 15, 82, 24, 64, 61, 67, 99, 50, 68, 25, 37, 32,  
27, 14, 91, 79, 15, 47, 48, 74, 88,  
64, 53, 77, 50, 24, 91, 87, 55, 60, 75, 91, 22, 47, 63, 81, 88, 26, 48, 69,  
59, 84, 77, 28, 36, 59, 74, 89, 73,  
91, 64, 55, 88, 90, 48, 73, 97, 98, 40, 93, 50, 78, 60, 44, 77, 82, 51, 53,  
65, 98, 59, 94, 91, 52, 44, 65, 85,  
72, 92, 49, 67, 58, 48, 62, 54, 89, 67, 58, 48, 85, 45, 77, 76, 81, 77]
```

Urutkan data tersebut menggunakan kedua jenis algoritma sorting: bubble sort dan selection sort.

## C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat 5 langkah pemrograman

1. Problem Statement

2. Mathematical Equation

2.1 Buble Sort

2.2 Slection Sort

3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)

3.1 Buble Sort

3.2 Slection Sort

4. Flowchart

4.1 Buble Sort

4.2 Slection Sort

5. Source Code

5.1 Buble Sort

5.2 Slection Sort

6. Tambahkan fungsi clock untuk menghitung waktu yang dibutuhkan untuk kedua lagoritma tersebut. Dengan data yang sama, manakah yang lebih cepat? Jawab dengan menunjukkan bukti screencapture hasil running dan perhitungan clock.

Jika program dijalankan dengan contoh data, maka luarannya adalah sebagai berikut.

Program Sorting

Nama : xxxxxxxxxxxxxxxxxxxxxx

NRM : 9999999999999999

DATA NILAI AWAL

```
85, 13, 99, 34, 71, 15, 82, 24, 64, 61,  
67, 99, 50, 68, 25, 37, 32, 27, 14, 91,  
79, 15, 47, 48, 74, 88, 64, 53, 77, 50,  
24, 91, 87, 55, 60, 75, 91, 22, 47, 63,  
81, 88, 26, 48, 69, 59, 84, 77, 28, 36,  
59, 74, 89, 73, 91, 64, 55, 88, 90, 48,  
73, 97, 98, 40, 93, 50, 78, 60, 44, 77,  
82, 51, 53, 65, 98, 59, 94, 91, 52, 44,  
65, 85, 72, 92, 49, 67, 58, 48, 62, 54,  
89, 67, 58, 48, 85, 45, 77, 76, 81, 77
```

-----

DATA DIURUTKAN DENGAN BUBLE SORT

...

Waktu yang dibutuhkan:

-----

DATA DIURUTKAN DENGAN SELECTION SORT

...

Waktu yang dibutuhkan:

-----

# MODUL 7

## DEFINING GRAPH



### Praktikum Pemrograman Komputer 115

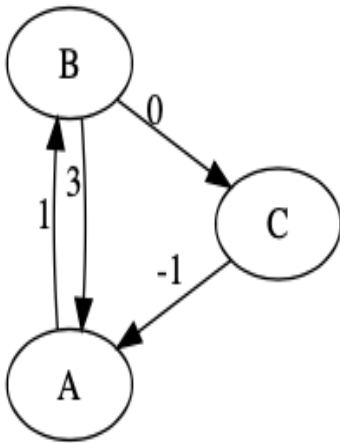
Kode Seksi: 1306600039

#### A. TUJUAN PEMBELAJARAN

1. Memahami graph sebagai koneksi antar titik
2. Menyimpan data graph dari jaringan ke dalam dictionary.

#### B. PERSOALAN

Perhatikan contoh hubungan tiga titik berikut.

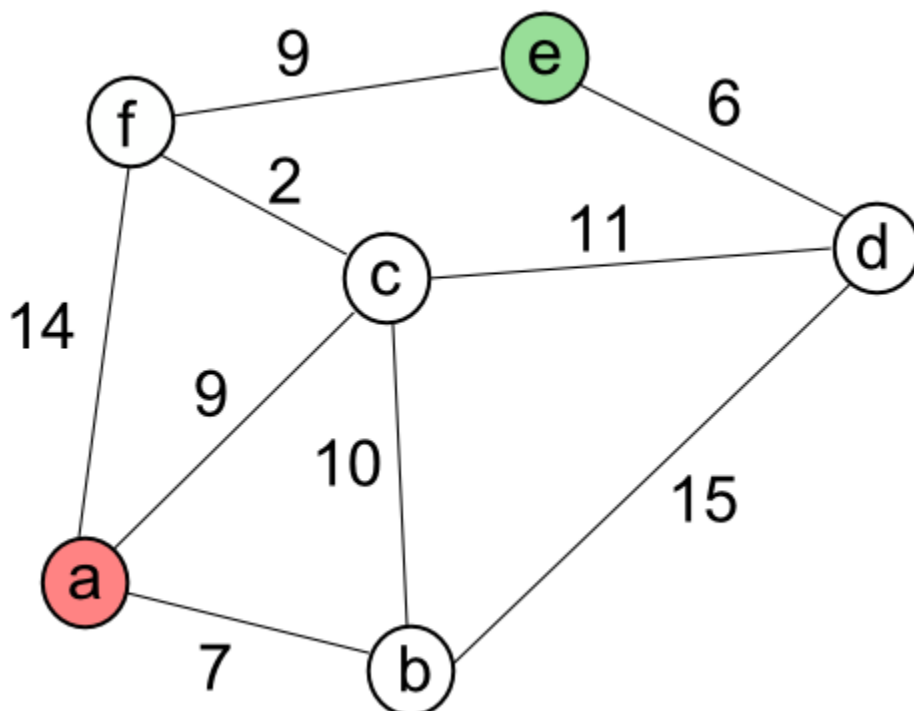


Jika dituliskan ke dalam source code dengan type dict, seperti berikut.

```
contoh_graph = {
    "A": {
        "B": 1,
    },
    "B": {
        "A": 3,
        "C": 0,
    },
    "C": {
        "A": -1,
    },
}
```

### C. PROGRAM DAN BENTUK LAPORAN YANG DIMINTA

Buat definisi graph dari gambar berikut. Anggap setiap titik memiliki **dua arah nilai**. Kemudian tentukan total jarak yang ditempuh dari titik a ke titik e dengan mempertimbangkan berbagai kemungkinan.



Laporan yang diminta

1. Problem Statement

2. Mathematical Equation

Penjumlahan element di dalam dict

3. Algoritma (algoritma merupakan pseudocode, seperti draft source code)

4. Flowchart >>> untuk laporan awal hanya diminta sampai dengan tahap ini.

5. Source Code

6. Tambahkan fungsi atau library yang dibutuhkan