



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUMINI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

NIM	71251193
Nama Lengkap	Cheila Zefanya Wibowo
Minggu ke / Materi	03/Flowchart dan Pseudocode

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2026**

BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

Algoritma

Algoritma adalah rangkaian terurut dengan langkah-langkah yang logis dan sistematis yang disusun untuk menyelesaikan masalah. Algoritma memiliki tujuan yaitu memberikan petunjuk tentang langkah-langkah yang mudah dipahami oleh nalar manusia. Dalam pemrograman, algoritma berfungsi sebagai instruksi untuk memproses input menjadi output. Algoritma biasanya harus memiliki instruksi yang jelas, memiliki input, terurut dengan jelas, dan selalu berhenti ketika sudah mencapai tujuan. Algoritma dapat mencegah dari terjadinya kesalahan logika pada program computer yang dikembangkan.

Ada 3 bentuk notasi algoritma, yaitu :

- Uraian Deskriptif
- Flowchart / Diagram Alir
- Pseudocode

Uraian Deskriptif

Contoh : menyelesaikan permasalahan menghitung luas dan keliling suatu persegi Panjang algoritma

Hitung_Luas_dan_Keliling_Persegi_Panjang

Deskripsi :

- Masukkan panjang dan lebar persegi Panjang
- Hitung luas Persegi Panjang dengan rumus $L = P \times L$

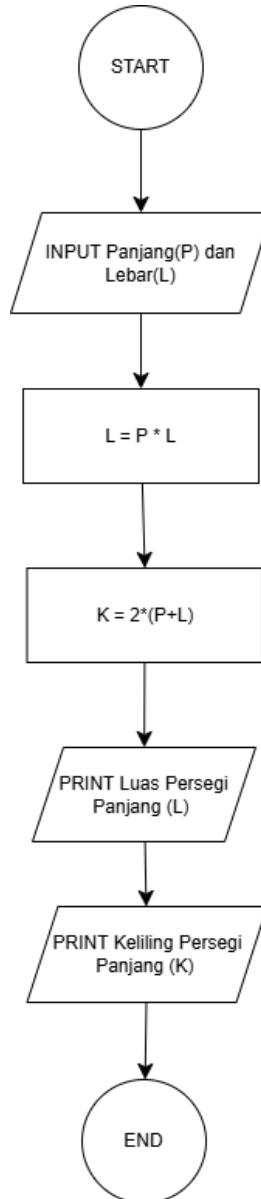
- Hitung Keliling Persegi Panjang dengan rumus $K = 2*(P+L)$
- Tampilkan Luas Persegi Panjang
- Tampilkan Keliling Persegi Panjang

Flowchart / Diagram Alir

Flowchart adalah diagram dari algoritma dalam suatu program yang menunjukkan langkah-langkah untuk menunjukkan alur dalam suatu program.

Contoh :

Menghitung Luas dan Keliling Persegi Panjang yang algoritma nya dalam bentuk diagram alir (flowchart)



Gambar 1.1 : Flowchart mencari luas dan keliling lingkaran

Flowchart sangat membantu untuk menggambarkan alur berjalannya program sebelum program itu jadi dan menolong dalam menganalisis alternatif lain dalam pengoperasian.

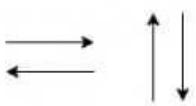
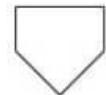
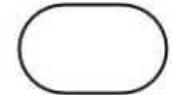
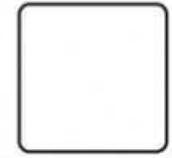
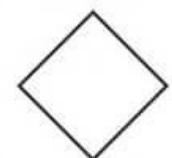
Kegunaan :

1. Untuk mendesain program.
2. Untuk mempresentasikan program.

Maka, flowchart harus dapat mempresentasikan komponen dalam bahasa pemrograman.

Notasi Flowchart

Notasi yang kita sering gunakan dalam pemrosesan flowchart adalah sebagai berikut.

	Flow Simbol yang digunakan untuk menggabungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan Connecting Line.
	On-Page Reference Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang sama.
	Off-Page Reference Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang berbeda.
	Terminator Simbol yang menyatakan awal atau akhir suatu program.
	Process Simbol yang menyatakan suatu proses yang dilakukan komputer.
	Decision Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya dan tidak.

Gambar 1.2 : Gambar Notasi Flowchart

	Input/output Simbol yang menyatakan proses input atau output tanpa tergantung peralatan.
	Manual Operation Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.
	Document Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak.
	Predefine Proses Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedure.
	Display Simbol yang menyatakan peralatan output yang digunakan.
	Preparation Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.

Gambar 1.3 : Gambar Notasi Flowchart

Notasi di atas memiliki fungsi yang berbeda-beda. Ada yang berfungsi sebagai awalan dan akhir, menunjukkan proses, mengabungkan antara symbol satu dengan yang lainnya, dan masih banyak lagi.

Pseudocode

pseudocode adalah notasi yang menyerupai notasi bahasa pemrograman Tingkat tinggi. Pseudocode itu deskripsi algoritma pemrograman yang menggunakan bahasa sehari-hari yang dikombinasikan dengan bahasa pemrograman. Algoritma memiliki beberapa struktur, yaitu:

1. Bagian Kepala (header)
2. Bagian Deklarasi (Definisi Variable)
3. Bagian Deskripsi (Rincian langkah)

Contoh :

```
Algoritma Luas_dan_keliling_persegi_panjang
{menghitung luas dan keliling persegi Panjang}
```

Deklarasi

```
float panjang, lebar, luas, keliling
```

Deskripsi

```
Read(panjang, lebar)
```

```
Luas ← panjang * lebar
```

```
Keliling ← 2 * (Panjang + lebar)
```

```
WRITE(Luas)
```

```
WRITE(Keliling)
```

Notasi Pseudocode

Berikut notasi yang sering digunakan dalam pseudocode:

1. **INPUT** : untuk menunjukkan proses memasukkan suatu isi variabel.
2. **OUTPUT** : untuk menunjukkan proses keluaran yang terjadi.
3. **WHILE** : untuk sebuah perulangan yang jumlah perulangannya belum pasti.
4. **FOR** : untuk sebuah perulangan yang jumlah perulangannya sudah pasti.
5. **REPEAT – UNTIL** : untuk sebuah perulangan yang memiliki kondisi akhir.
6. **IF – THEN – ELSE** : untuk mengambil sebuah keputusan dari beberapa kondisi.

Sumber : Materi 03 flowchart dan pseudocode (eclass)

BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

Link GitHub : https://github.com/chei-cell/71251193_cheila-.git

SOAL 1

Pseudocode :

Algoritma_Bilangan_Prima

Deklarasi

angka, i : integer

Deskripsi

START

INPUT angka

IF angka <= 1 THEN

 PRINT "tidak bilangan prima"

ELSE

 i = 2

 WHILE i <= angka-1 DO

 IF angka MOD i = 0 THEN

 PRINT "tidak bilangan prima"

 END

 ENDIF

 i = i + 1

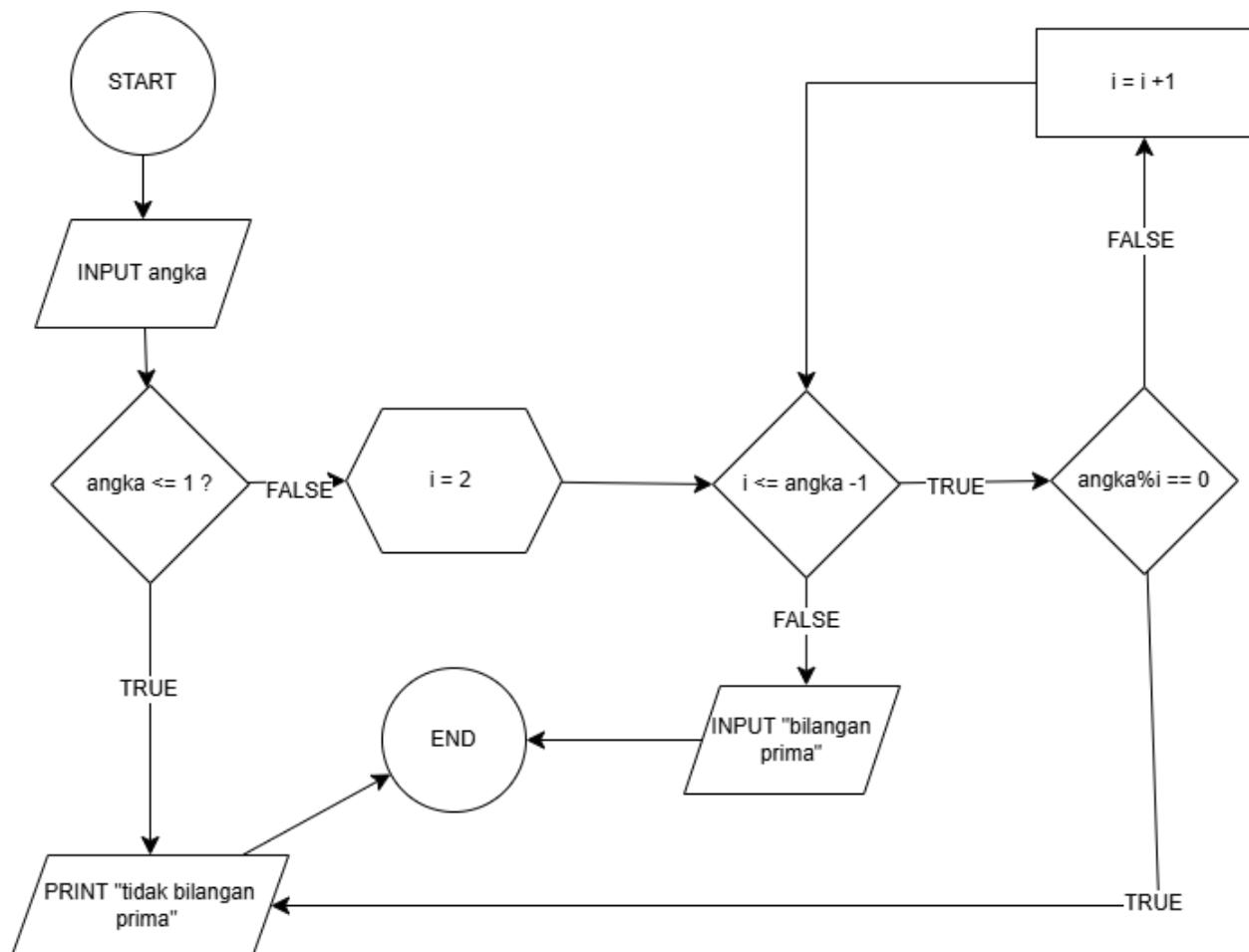
ENDWHILE

```
PRINT "bilangan prima"  
ENDIF  
END
```

Langkah – langkah :

1. Mulai program dengan start
2. Masukkan bilangan yang mau diperiksa dengan INPUT angka
3. Cek kondisi apakah angka ≤ 1
4. Jika angka ≤ 1 tampilkan “tidak bilangan prima” dan program selesai (bilangan prima harus lebih dari 1)
5. Jika angka > 1 , set $i = 2$
6. Melakukan perulangan menggunakan while karena jumlah perulangan yang belum pasti.
7. $i \leq \text{angka} - 1$, jika salah akan langsung menghasilkan “bilangan prima”. Jika benar akan lanjut memeriksa dengan **angka MOD i = 0**
8. jika angka MOD i = 0 maka akan menampilkan “tidak bilangan prima”.
9. Jika tidak naikkan nilai i dengan cara $i = i + 1$
10. Kembali cek ke $i \leq \text{angka}-1$
11. Setelah mendapatkan bilangan prima program akan berakhir (END)

Flowchart :



SOAL 2

Pseudocode :

Algoritma Jenis_Akar_Persamaan_Kuadrat

{Menentukan jenis akar dari persamaan $ax^2 + bx + c = 0$ }

Deklarasi

a, b, c : float

Deskripsi

```

START
INPUT a, b, c
D ← (b*b) – (4*a*c)

IF D>0 THEN
    PRINT ("Akar Real dan Berlainan")
ELSE D=0 THEN
    PRINT ("Akar Real dan Kembar")
ELSE
    PRINT ("Akar Imaginer dan Tidak Real")
ENDIF
END

```

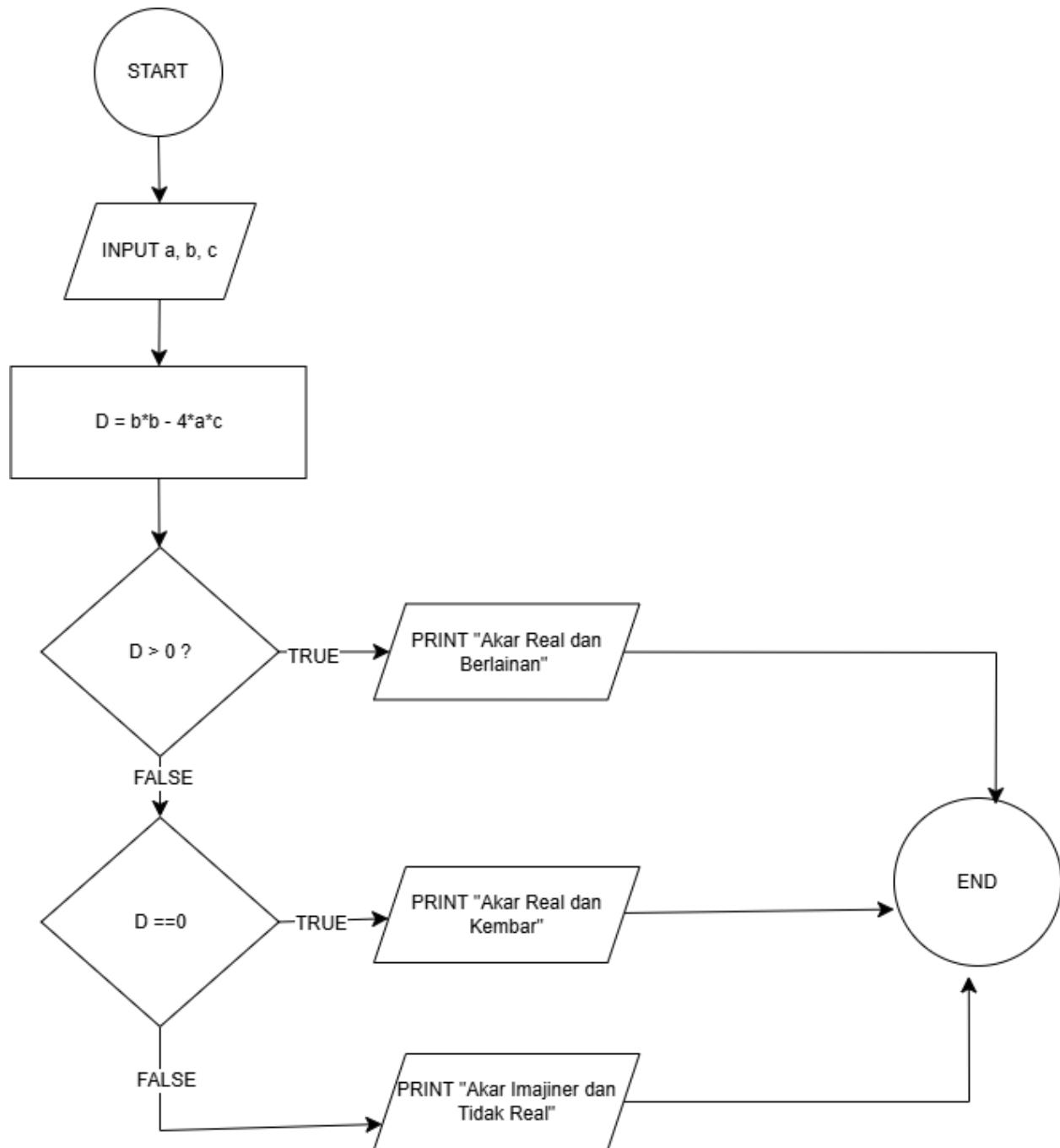
Penjelasan :

1. Untuk memulai algoritma kita harus menggunakan start.
2. Masukkan variabel yaitu nilai a,b,c agar bisa mencari nilai diskriminan dari persamaan kuadrat.
3. Untuk mencari diskriminan kita menggunakan rumus yaitu $D = b^2 - 4ac$
4. Untuk menentukan jenis akar, kita perlu membuat beberapa kondisi seperti pada pseudocode diatas
5. Kita menggunakan if then else karena harus memilih satu kondisi dari beberapa kemungkinan.
6. Kita periksa nilai $D > 0$ maka akan menampilkan "Akar Real dan Berlainan", jika $D = 0$ maka akan menampilkan "Akar Real dan Kembar", jika $D < 0$ maka akan menampilkan "Akar Imaginer dan Tidak Real"

7. Jika tidak menggunakan if then else program tidak bisa mengecek kondisi dia akan berjalan lurus tanpa memilih kondisi

8. END untuk mengakhiri sebuah program

Flowchart :



SOAL 3

Pseudocode :

Algoritma Hitung_Faktorial

Deklarasi

```
angka : integer  
i : integer  
faktorial : integer
```

Deskripsi

```
START  
SET faktorial = z  
INPUT angka  
IF angka<0 THEN  
    PRINT ("tidak boleh negatif")  
ELSE  
    i = 1  
    WHILE i <= angka DO  
        faktorial = faktorial * i  
        i = i+1  
    ENDWHILE  
    PRINT ("Hasil Faktorial")  
ENDIF  
END
```

Langkah-langkah :

1. Mulai program selalu dengan start
2. Lalu, kita set nilai awal yaitu faktorial = 1 (karena faktorial selalu mulai dari 1)
3. Kita input angka

4. Lalu kita menggunakan if untuk mengecek apakah angka < 0, jika angka < 0 maka tidak bisa menghasilkan faktorial dan akan mengeluarkan output "tidak boleh negatif"
5. Lalu kita set untuk angka>0 yaitu i = 1
6. Setelah itu kita menggunakan while untuk cek kondisi perulangan, kenapa menggunakan while karena jumlah perulangan nya belum pasti.
7. WHILE i <= angka, jika salah maka akan langsung menghasilkan "Hasil faktorial". Tetapi, jika benar maka akan langsung menghitung faktorial = faktorial * i
8. Setelah itu, akan menaikkan nilai i dengan i = i + 1
9. Setelah itu akan kembali mengecek kondisi ke langkah 7
10. Program berhenti jika sudah menghasilkan output dengan menggunakan END

Flowchart :

