



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2025/2026

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI 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	71251210
Nama Lengkap	Gabriel Ekklesia
Minggu ke / Materi	03 / Flowchart dan Pseudocode

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

BAGIAN 1: MATERI MINGGUINI (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.

MATERI 1

Algoritma

Algoritma adalah rangkaian langkah-langkah logis, sistematis, dan terstruktur yang dirancang untuk menyelesaikan suatu permasalahan. Algoritma berfungsi sebagai pedoman dalam menentukan urutan proses penyelesaian masalah sehingga solusi yang dihasilkan menjadi jelas, tepat, dan mudah dipahami. Dalam bidang pemrograman, algoritma berperan sebagai dasar dalam pembuatan program komputer.

Pemahaman yang baik terhadap algoritma sangat penting karena dapat membantu mengurangi kemungkinan terjadinya kesalahan logika sejak tahap perancangan program. Dengan algoritma yang tersusun secara benar, proses pengembangan program menjadi lebih terarah, efisien, dan sistematis.

Penulisan (Notasi Algoritma) Ada tiga macam bentuk notasi algoritma antara lain:

1. Uraian deskriptif
2. Flowchart / Diagram Alir
3. Pseudocode

MATERI 2

Uraian Deskriptif

menyelesaikan permasalahan menghitung luas dan keliling suatu lingkaran Algoritma Hitung_Luas_dan_Keliling_Lingkaran.

Deskripsi:

1. Masukkan jari-jari lingkaran (r).

2. Hitung luas lingkaran dengan rumus $L = \pi * r^2$.
3. Hitung keliling lingkaran dengan rumus $K = 2 * \pi * r$.
4. Tampilkan luas lingkaran.
5. Tampilkan keliling lingkaran.

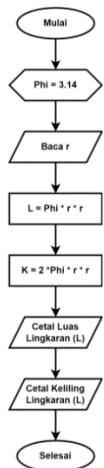
MATERI 3

Flowchart / Diagram Ali

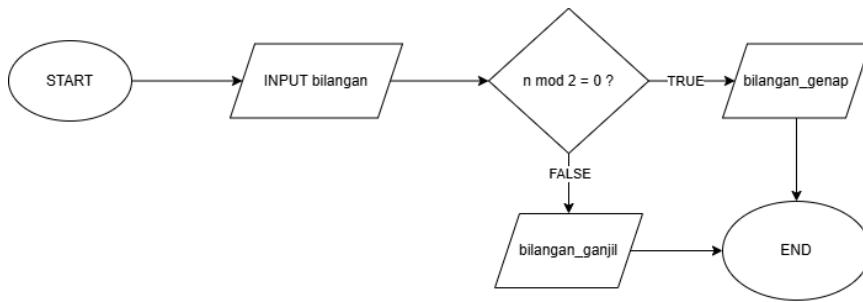
Flowchart adalah representasi visual dari suatu algoritma yang digambarkan dalam bentuk diagram alir menggunakan simbol-simbol tertentu yang saling terhubung oleh garis panah. Diagram ini menunjukkan urutan langkah-langkah proses secara sistematis serta menggambarkan arah aliran logika dari suatu program atau penyelesaian masalah.

Flowchart berfungsi untuk mempermudah pemahaman terhadap alur kerja suatu sistem atau program sebelum diimplementasikan ke dalam bahasa pemrograman. Dengan menggunakan flowchart, perancang program dapat melihat struktur proses secara jelas, mengidentifikasi kemungkinan kesalahan logika.

Menghitung luas dan keliling lingkaran yang algoritmanya dinotasikan dalam bentuk diagram alir (flowchart).



Contoh flowchart dalam bentuk ganjil genap



Flowchart menolong analis dan programmer dalam memecahkan suatu masalah menjadi bagian-bagian yang lebih kecil dan terstruktur. Dengan memvisualisasikan langkah-langkah proses secara sistematis.

Kegunaan:

1. Untuk mendesain program.
2. Untuk merepresentasikan program.

Flowchart harus dapat merepresentasikan komponen-komponen dalam bahasa pemrograman.

MATERI 4

Notasi Flowchart

notasi dalam flowchart memiliki arti yang berbeda-beda

	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.
	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 procedure.
	Display Simbol yang menyatakan peralatan output yang digunakan.
	Preparation Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.

Notasi-notasi pada flowchart memiliki jenis dan fungsi yang berbeda-beda sesuai dengan perannya dalam menggambarkan alur proses. Beberapa notasi berfungsi untuk

menghubungkan satu simbol dengan simbol lainnya, seperti flowline (garis alir) yang menunjukkan arah aliran proses, serta on-page reference dan off-page reference yang digunakan untuk menghubungkan bagian flowchart yang terpisah.

Selain itu, terdapat notasi yang berfungsi untuk menunjukkan suatu proses yang sedang berlangsung, yaitu simbol process, yang menggambarkan langkah pengolahan data atau instruksi yang dijalankan dalam sistem. Di samping itu, terdapat pula notasi yang digunakan untuk menerima masukan (input) dan menampilkan keluaran (output), yaitu simbol input/output, yang berperan dalam proses interaksi antara pengguna dan sistem.

MATERI 5

Pseudocode

Pseudocode merupakan notasi yang menyerupai bahasa pemrograman tingkat tinggi, seperti Bahasa C dan Python, namun tidak terikat pada aturan sintaksis tertentu. Pseudocode digunakan untuk menggambarkan langkah-langkah penyelesaian suatu masalah secara logis dan sistematis sehingga mudah dipahami. Struktur algoritma dibagi jadi beberapa bagian:

1. Bagian kepala (header).
2. Bagian Deklarasi (definisi variable).
3. Bagian Deskripsi (rincian langkah).

```
Algoritma Luas_persegi_panjang
{Menghitung sebuah luas persegi panjang

    apabila panjang dan lebar persegi panjang
    tersebut diberikan}

Deklarasi
{Definisi nama peubah/variabel}
float panjang, lebar, luas

Deskripsi
READ (panjang,lebar)          # bisa juga : INPUT
luas <- panjang * lebar
WRITE (Luas)                  # bisa juga : OUTPUT
```

Contoh pseudocode dari bilangan ganjil genap

IF Bil Mod 2 = 0 THEN

```
"Output Genap"  
Else  
    "Output Ganjil"  
END IF
```

MATERI 6

Notasi Pseudocode

Berikut ini notasi yang sering digunakan dalam pseudocode:

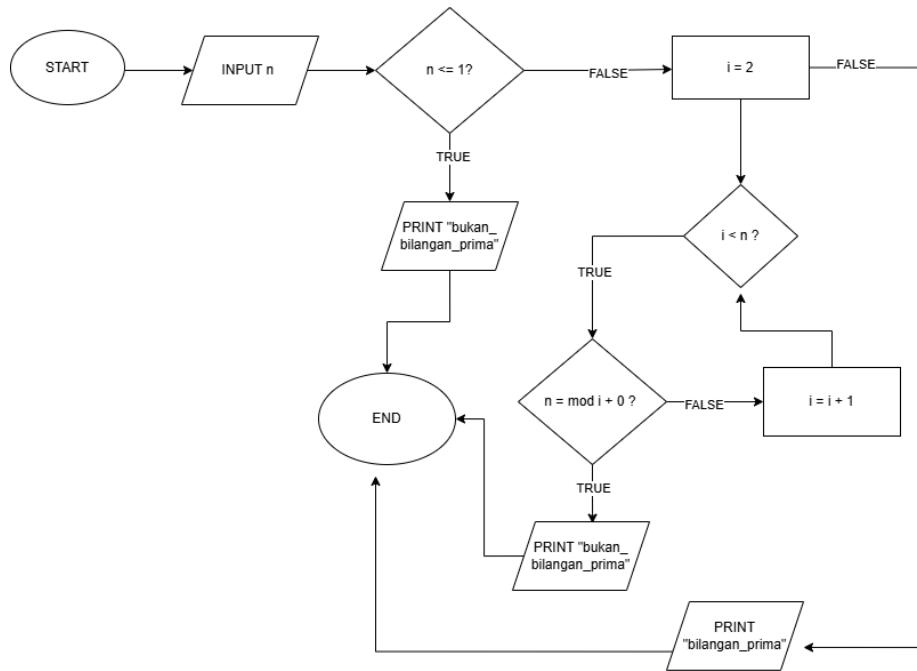
1. INPUT -> Digunakan untuk menunjukan proses memasukan suatu isi variabel.
2. OUTPUT -> Digunakan untuk menunjukan proses keluaran yang terjadi.
3. WHILE -> Digunakan untuk sebuah perulangan yang memiliki iterasi awali.
4. FOR -> Digunakan untuk sebuah perulangan perhitungan iterasi.
5. REPEAT - UNTIL -> Digunakan untuk sebuah perulangan yang memiliki kondisi akhir.
6. IF – THEN – ELSE -> Digunakan untuk mengambil sebuah keputusan dari beberapa kondisi.

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.

SOAL 1

FLOWCHART



Flowchart tersebut menggambarkan algoritma untuk menentukan apakah suatu bilangan merupakan bilangan prima atau bukan. Proses dimulai dengan menerima input berupa bilangan n dari pengguna. Selanjutnya dilakukan pengecekan awal, yaitu apakah nilai $n \leq 1$. Jika kondisi tersebut terpenuhi, maka bilangan dinyatakan bukan bilangan prima dan proses dihentikan.

Apabila $n > 1$, algoritma melanjutkan dengan menginisialisasi variabel pembagi $i = 2$. Kemudian dilakukan proses perulangan untuk memeriksa apakah terdapat bilangan antara 2 hingga kurang dari n yang dapat membagi n secara habis. Pada setiap iterasi, dilakukan pengujian apakah $n \bmod i = 0$. Jika kondisi tersebut benar, maka n memiliki pembagi selain 1 dan dirinya sendiri sehingga dinyatakan bukan bilangan prima dan proses berakhir. Jika tidak habis dibagi, nilai i ditambah satu dan proses pengujian diulang.

Apabila seluruh nilai pembagi telah diperiksa dan tidak ditemukan pembagi yang memenuhi kondisi tersebut, maka bilangan dinyatakan sebagai bilangan prima. Dengan demikian, flowchart ini menunjukkan proses penentuan bilangan prima menggunakan metode pengujian pembagian berulang secara sistematis.

Pseudocode

ALGORITMA Cek_Bilangan_Prima

DEKLARASI

n, i : integer

DESKRIPSI

INPUT n

IF n <= 1 THEN

 OUTPUT "Bukan bilangan prima"

ELSE

 i ← 2

 WHILE i < n DO

 IF n MOD i = 0 THEN

 OUTPUT "Bukan bilangan prima"

 END

 ENDIF

 i ← i + 1

 ENDWHILE

 OUTPUT "Bilangan prima"

ENDIF

END

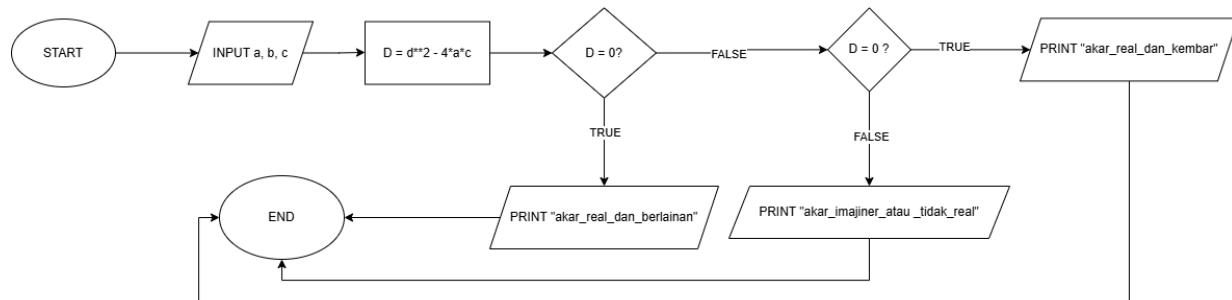
Algoritma Cek Bilangan Prima digunakan untuk menentukan apakah suatu bilangan termasuk bilangan prima. Proses dimulai dengan menerima input bilangan n, kemudian dilakukan pengecekan awal. Jika n <= 1, maka bilangan dinyatakan bukan bilangan prima.

Jika n > 1, algoritma menginisialisasi i = 2 sebagai pembagi awal. Selanjutnya dilakukan perulangan untuk memeriksa apakah terdapat bilangan antara 2 hingga kurang dari n

yang dapat membagi n secara habis. Jika ditemukan pembagi yang memenuhi kondisi $n \bmod i = 0$, maka bilangan dinyatakan bukan bilangan prima. Jika tidak ditemukan pembagi hingga perulangan selesai, maka bilangan dinyatakan sebagai bilangan prima.

SOAL 2

FLOWCHART



Flowchart tersebut menggambarkan algoritma untuk menentukan jenis akar-akar persamaan kuadrat berdasarkan nilai diskriminan. Proses dimulai dengan menerima input berupa tiga koefisien persamaan kuadrat, yaitu a, b, dan c dari bentuk umum persamaan kuadrat $ax^2+bx+c=0$. Setelah itu, sistem menghitung nilai diskriminan menggunakan rumus $D = b^2 - 4ac$.

Selanjutnya dilakukan proses pengambilan keputusan berdasarkan nilai diskriminan yang diperoleh. Apabila nilai D lebih besar dari nol, maka persamaan kuadrat memiliki dua akar real yang berlainan. Jika nilai D sama dengan nol, maka persamaan kuadrat memiliki dua akar real yang sama atau kembar. Sementara itu, jika nilai D kurang dari nol, maka persamaan kuadrat tidak memiliki akar real atau memiliki akar imajiner.

Setelah jenis akar persamaan kuadrat ditentukan dan ditampilkan sebagai keluaran, proses algoritma diakhiri. Flowchart ini menunjukkan alur penyelesaian masalah secara sistematis dalam menentukan karakteristik akar persamaan kuadrat berdasarkan nilai diskriminannya.

Pseudocode

ALGORITMA Jenis_Akar_Persamaan_Kuadrat

DEKLARASI

a, b, c : real

D : real

DESKRIPSI

INPUT a, b, c

$D \leftarrow b^2 - 4ac$

IF $D > 0$ THEN

 OUTPUT "akar real dan berlainan"

ELSE

 IF $D = 0$ THEN

 OUTPUT "akar real dan kembar"

 ELSE

 OUTPUT "akar imajiner atau tidak real"

 ENDIF

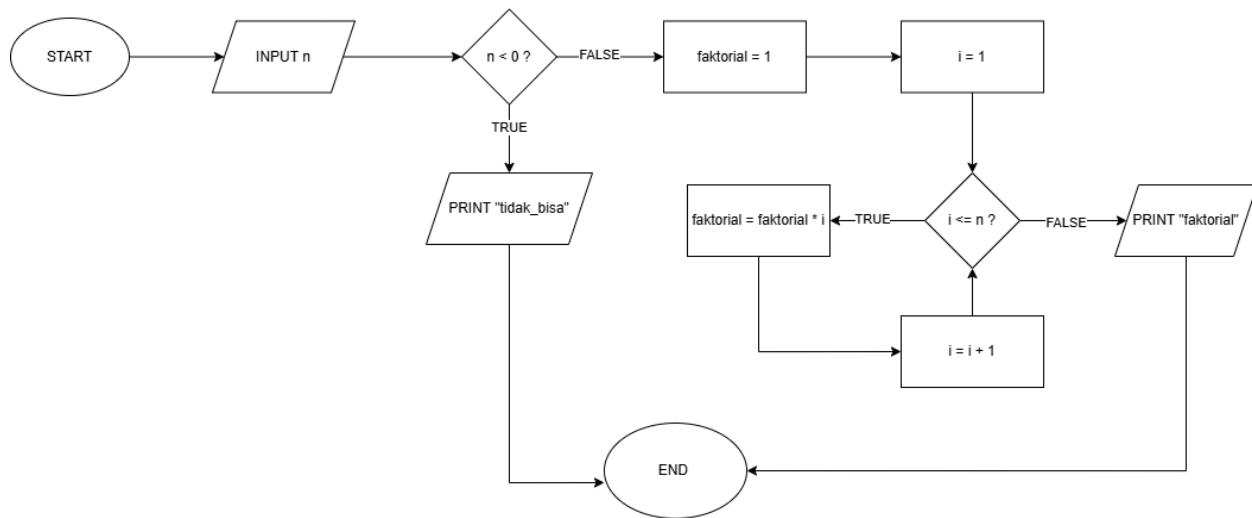
ENDIF

END

Algoritma Jenis Akar Persamaan Kuadrat digunakan untuk menentukan jenis akar dari suatu persamaan kuadrat berdasarkan nilai diskriminan. Proses dimulai dengan menerima input koefisien a, b, dan c, kemudian menghitung nilai diskriminan dengan rumus $D = b^2 - 4ac$.

Selanjutnya dilakukan pengambilan keputusan berdasarkan nilai D. Jika $D > 0$, maka persamaan memiliki dua akar real yang berlainan. Jika $D = 0$, maka persamaan memiliki akar real kembar. Jika $D < 0$, maka persamaan tidak memiliki akar real atau memiliki akar imajiner. Setelah jenis akar ditentukan, proses algoritma selesai.

FLOWCHART



Proses dimulai dengan menerima input bilangan n. Selanjutnya dilakukan pengecekan apakah $n < 0$. Jika kondisi tersebut benar, maka program menampilkan pesan "*tidak bisa*" karena faktorial tidak didefinisikan untuk bilangan negatif, kemudian proses dihentikan.

Jika $n \geq 0$, algoritma menginisialisasi nilai faktorial = 1 dan i = 1. Kemudian dilakukan proses perulangan selama $i \leq n$. Pada setiap iterasi, nilai faktorial diperbarui dengan mengalikan nilai faktorial sebelumnya dengan i, lalu nilai i ditambah satu. Proses ini berulang hingga kondisi perulangan tidak terpenuhi.

Setelah perulangan selesai, program menampilkan hasil faktorial dan proses diakhiri. Flowchart menunjukkan perhitungan faktorial menggunakan metode perulangan secara sistematis.

Pseudocode

ALGORITMA Hitung Faktorial

DEKLARASI

n, i : integer

faktorial : integer

DESKRIPSI

```
INPUT n  
IF n < 0 THEN  
    OUTPUT "tidak bisa"  
ELSE  
    faktorial ← 1  
    i ← 1  
    WHILE i ≤ n DO  
        faktorial ← faktorial * i  
        i ← i + 1  
    ENDWHILE  
    OUTPUT faktorial  
ENDIF  
END
```

Algoritma Hitung Faktorial digunakan untuk menghitung nilai faktorial dari suatu bilangan bulat n . Proses dimulai dengan menerima input n , kemudian dilakukan pengecekan apakah $n < 0$. Jika kondisi tersebut terpenuhi, program menampilkan pesan "*tidak bisa*" karena faktorial tidak didefinisikan untuk bilangan negatif.

Jika $n \geq 0$, algoritma menginisialisasi faktorial = 1 dan $i = 1$. Selanjutnya dilakukan perulangan selama $i \leq n$. Pada setiap iterasi, nilai faktorial dikalikan dengan i , kemudian nilai i ditambah satu. Setelah perulangan selesai, program menampilkan hasil faktorial dan proses berakhir.

LINK GITHUB: https://github.com/gilbetjuliesa-crypto/71251210_Gabriel_Ekklesia.git