



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 KULIAHINI.

NIM	<71251177>
Nama Lengkap	<NEO KURNIAWAN PUTRA SANTOSA>
Minggu ke / Materi	03 / Flowchart and Pseudocode

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

BAGIAN 1: MATERI MINGGUINI (40%)

Algoritma

Algoritma adalah rangkaian yang terurut dan langkah - langkah logis dan sistematis yang disusun untuk menyelesaikan suatu masalah. Tujuan algoritma adalah memberikan petunjuk langkah logika penyelesaian masalah dalam bentuk yang mudah di pahami oleh manusia sebagai acuan yang membantu dalam mengembangkan program computer. Penulisan atau notasi algoritma ada tiga yaitu uraian deskriptif, Flowchart, Pseudocode.

Uraian Deskriptif

Uraian Deskriptif adalah cara penulisan algoritma yang menggunakan kalimat deskriptif

contohnya : menyelesaikan permasalahan menghitung luas keliling suatu 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.

Flowchart / Diagram Alir

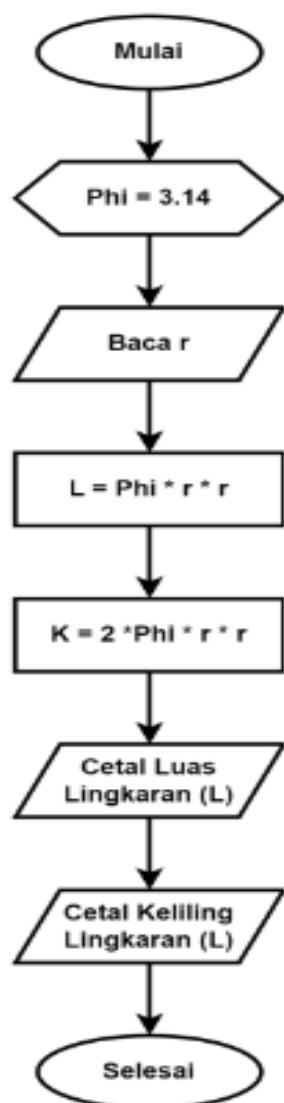
Flowchart atau Diagram alir adalah Gambaran dalam bentuk diagram alir dari Algoritma – algoritma

dalam suatu program, Flowchart atau Diagram alir juga menggunakan simbol – simbol standart untuk

memudahkan pembaca memahami alur yang digambarkan, flowchart atau diagram alir digunakan pada

bidang pemrograman, bisnis, dan sistem computer. Flowchart juga dapat membantu untuk menyelesaikan masalah kedalam segmen yang lebih kecil dan menolong dalam menganalisis alternatif

dalam pengoperasian dalam mendesain program dan untuk merepresentasikan program.

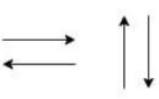
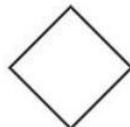


Ini contoh dari flowchart dari menghitung luas dan keliling sebuah lingkaran.

Notasi pada Flowchart

Flowchart memiliki berbagai notasi yang masing-masing memiliki arti berbeda. Flow menghubungkan simbol, on-page reference dan off-page reference menyambungkan proses pada halaman yang sama atau berbeda. Terminator menandai awal atau akhir program, process menunjukkan langkah yang dilakukan komputer, dan decision menyatakan percabangan dengan dua kemungkinan jawaban.

Simbol input/output menunjukkan proses masukan atau keluaran, manual operation untuk proses yang tidak dilakukan komputer, document untuk input atau output berbentuk dokumen, predefined process untuk bagian proses tertentu, display untuk menampilkan hasil, dan preparation untuk tahap persiapan atau pemberian nilai awal.

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

Pseudocode

Pseudocode adalah cara menuliskan algoritma dengan gaya yang mirip bahasa pemrograman tingkat tinggi seperti C, C++, atau Python, namun tidak mengikuti aturan sintaks yang kaku. Pseudocode digunakan untuk menjelaskan alur logika program secara runtut dan mudah dipahami sebelum diterapkan ke dalam bahasa pemrograman yang sebenarnya. Karena tidak terikat aturan baku, penulisannya lebih fokus pada kejelasan langkah-langkah daripada ketepatan struktur bahasa.

Secara umum, pseudocode memiliki beberapa bagian utama. Bagian pertama adalah header, yang memuat nama algoritma beserta penjelasan singkat tentang tujuannya. Selanjutnya terdapat bagian deklarasi, yaitu tempat mendefinisikan variabel, konstanta, atau data yang akan digunakan. Terakhir adalah bagian deskripsi, yang berisi langkah-langkah penyelesaian masalah secara terstruktur dari awal hingga menghasilkan output.

Contoh pseudocode menghitung luas lingkaran:

Algoritma_Luas_Lingkaran

{menghitung luas lingkaran apabila jari-jari diberikan}

Deklarasi

Float jari_jari, luas

Const pi = 3.14

Deskripsi

READ(jari_jari)

luas ← pi * jari_jari * jari_jari

WRITE(luas)

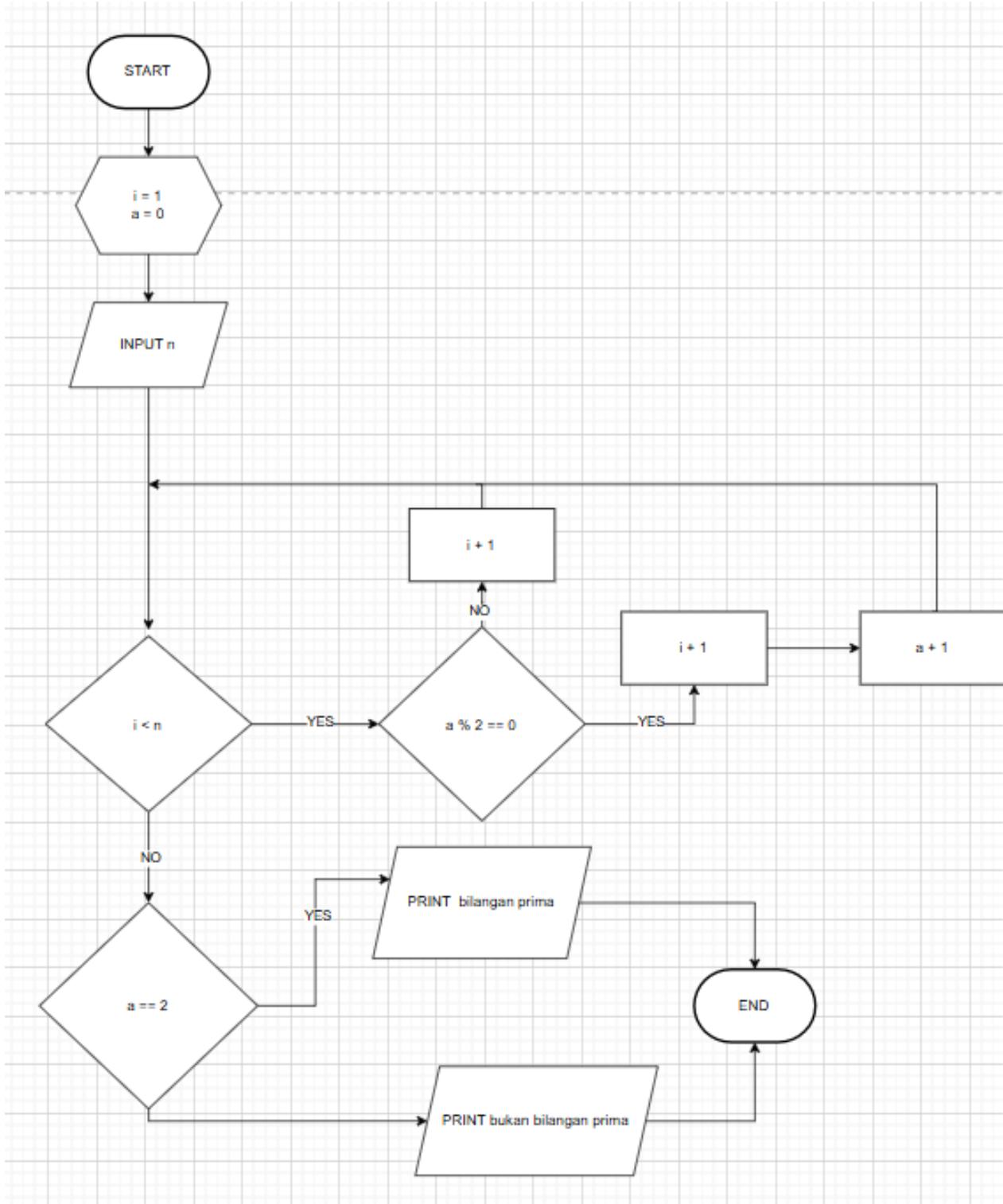
Notasi Pseudocode

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%)

Program Bilangan Prima



Pseudocode

```
#Deklarasi  
var int i,a,n  
  
#Deskripsi  
i <- 1  
a <- 0  
input n  
  
WHILE i < n Do  
if n % i == 0 Then  
a <- a + 1  
END IF  
i <- i + 1  
END WHILE  
if a == 2 Then  
print "Bilangan Prima"  
ELSE  
print "Bukan Bilangan Prima"  
END IF
```

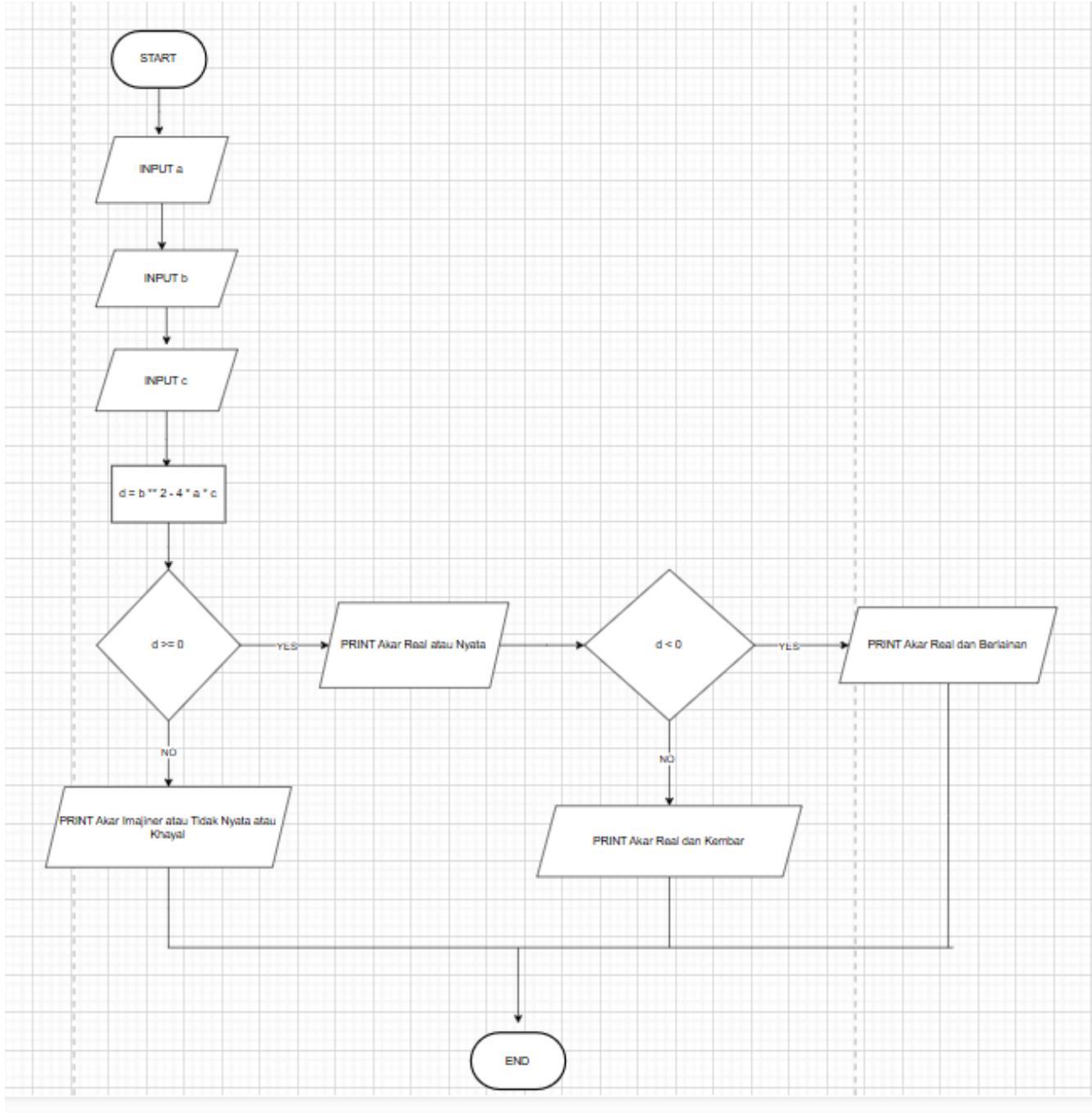
Penjelasan:

1. Pertama mendeklarasikan tiga variabel integer i, a, dan n.
2. Variabel i diinisialisasi dengan nilai 1 sebagai penghitung awal.
3. Variabel a diinisialisasi dengan nilai 0 sebagai penghitung jumlah faktor.
4. Pengguna input memasukkan nilai n.
5. Gunakan while untuk perulangan selama i masih lebih kecil dari n.
6. Pada setiap perulangan, program mengecek apakah n habis dibagi i menggunakan operasi modulus.
7. Jika $n \% i == 0$, maka variabel a ditambah 1 karena ditemukan faktor.
8. Setelah pengecekan, nilai i dinaikkan 1 untuk mencoba pembagi berikutnya.
9. Perulangan berlanjut sampai kondisi $i < n$ tidak terpenuhi.
10. Setelah perulangan selesai, program mengecek apakah jumlah faktor (a) sama dengan 2.

11. Jika $a == 2$, program menampilkan "Bilangan Prima" Jika tidak, program menampilkan "Bukan Bilangan Prima".
 12. Program selesai.

Source code: <https://github.com/NeoZeon30/pralpro-meet3>

Program Akar Real atau Kembar



Pseudocode

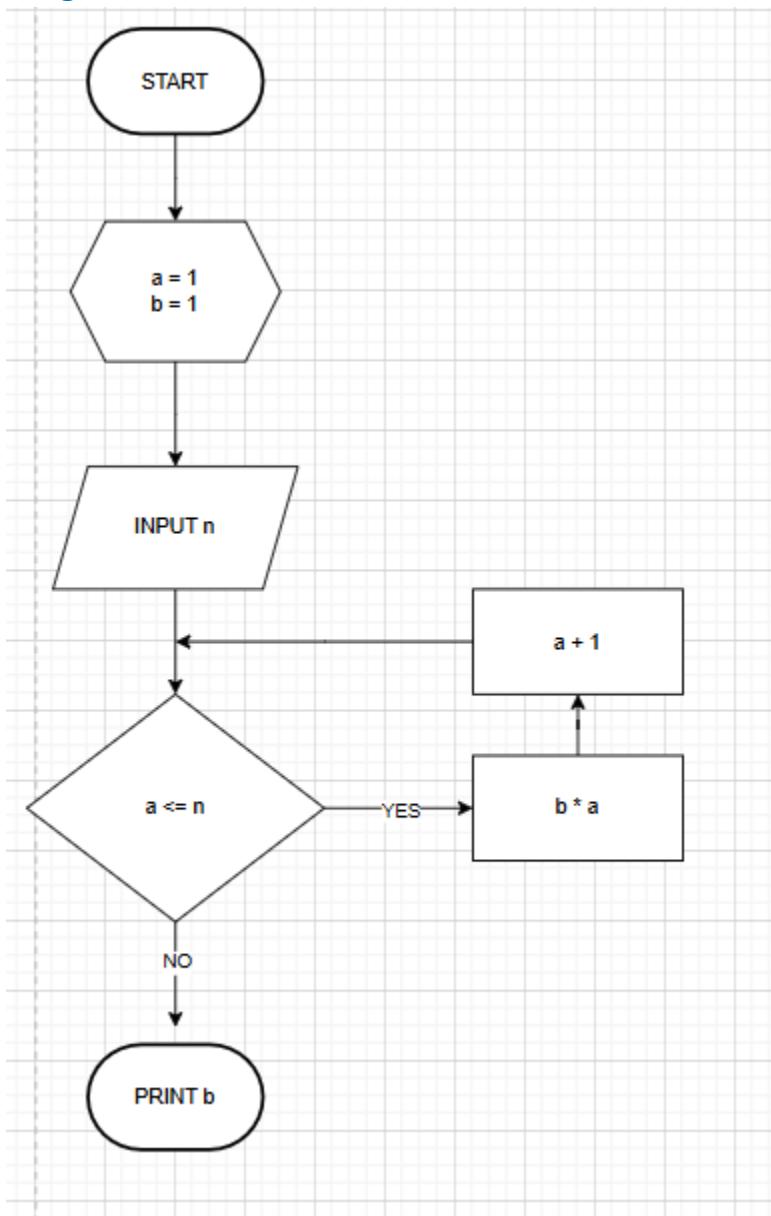
```
#Deklarasi  
  
var int a,b,c  
  
#Deskripsi  
  
input a,b,c  
  
d <- b ** 2 - 4 * a * c  
  
if d >= 0 THEN  
  
print "Akar Real atau Nyata"  
  
if d > 0 THEN  
  
print "Akar Real dan Berlainan"  
  
else d = 0 THEN  
  
print "Akar Real dan Kembar"  
  
else d < 0 THEN  
  
print "Akar Imajiner atau Tidak Real atau Khayal."  
  
END IF
```

Penjelasan:

1. Mendeklarasikan tiga variabel integer a, b, dan c sebagai koefisien persamaan kuadrat.
2. Input untuk Pengguna memasukkan nilai a, b, dan c.
3. Proses menghitung diskriminan dengan rumus $d <- b^{**} 2 - 4 * a * c$.
4. Proses mengecek apakah nilai diskriminan d lebih besar atau sama dengan 0.
5. Jika $d \geq 0$, program menampilkan pesan "Akar Real atau Nyata".
6. kemudian mengecek apakah $d > 0$.
7. Jika benar, program menampilkan "Akar Real dan Berlainan" (dua akar real berbeda).
8. Jika tidak, program mengecek apakah $d = 0$.
9. Jika $d = 0$, program menampilkan "Akar Real dan Kembar" (akar real sama).
10. Jika kondisi sebelumnya tidak terpenuhi dan $d < 0$, program menampilkan "Akar Imajiner atau Tidak Real atau Khayal."
11. Program selesai.

source code: <https://github.com/NeoZeon30/pralpro-meet3>

Program Faktorial



Pseudocode

```
# Deklarasi
```

```
var int a,b,n
```

```
#Deskripsi
```

```
a <- 1
```

```
b <- 1
```

```
INPUT n
```

```
WHILE a <= n DO
    b <- b * a
    a <- a + 1
END WHILE
OUTPUT b
```

Penjelasan:

1. Dimulai dengan mendeklarasikan tiga variabel integer a, b, dan n.
2. Variabel a diinisialisasi dengan nilai 1. Variabel ini akan berfungsi sebagai penghitung perulangan.
3. Variabel b diinisialisasi dengan nilai 1. Variabel ini akan menyimpan hasil perkalian (faktorial).
4. Input untuk pengguna memasukkan nilai n, yaitu bilangan yang akan dihitung faktorialnya.
5. Proses memulai perulangan selama nilai a masih kurang dari atau sama dengan n.
6. Pada setiap perulangan, nilai b dikalikan dengan a. Ini membangun hasil faktorial secara bertahap.
7. Setelah itu, nilai a ditambah 1 untuk melanjutkan ke angka berikutnya.
8. Perulangan terus berjalan sampai kondisi a <= n tidak terpenuhi.
9. Setelah perulangan selesai, program menampilkan nilai b sebagai hasil akhir.
10. Program selesai.

Source code: <https://github.com/NeoZeon30/pralpro-meet3>