**Pseudocode**

1. **Gerak Jatuh Bebas**

// Program untuk Menghitung Waktu Reaksi Maksimum

MULAI

// 1. Deklarasi Variabel

DEKLARASIKAN variabel numerik: tinggi\_pohon, tinggi\_andi\_cm, tinggi\_andi\_m, gravitasi, jarak\_jatuh, waktu\_reaksi

// 2. Inisialisasi Nilai (Input)

SET tinggi\_pohon = 10

SET tinggi\_andi\_cm = 160

SET gravitasi = 9.8

// 3. Proses Perhitungan

// Konversi tinggi Andi dari cm ke m

SET tinggi\_andi\_m = tinggi\_andi\_cm / 100

// Hitung jarak yang ditempuh kelapa hingga mencapai kepala Andi

SET jarak\_jatuh = tinggi\_pohon - tinggi\_andi\_m

// Hitung waktu yang dibutuhkan menggunakan rumus gerak jatuh bebas

SET waktu\_reaksi = AKAR\_KUADRAT((2 \* jarak\_jatuh) / gravitasi)

// 4. Tampilkan Hasil (Output)

TULISKAN "Waktu reaksi maksimum yang dimiliki Andi adalah: " + waktu\_reaksi + " detik"

SELESAI

// Program untuk Membandingkan Solusi Analitik dan Numerik Gerak Jatuh Bebas

MULAI

// Inisialisasi Konstanta dan Parameter

DEKLARASIKAN variabel numerik: tinggi\_awal, gravitasi, dt, t, v\_numerik, h\_numerik

SET tinggi\_awal = 10

SET gravitasi = 10

SET dt = 0.1 // Langkah waktu untuk simulasi numerik (semakin kecil, semakin akurat)

// -- A. Perhitungan Analitik --

DEKLARASIKAN array: waktu\_analitik, tinggi\_analitik

SET waktu\_total = AKAR\_KUADRAT((2 \* tinggi\_awal) / gravitasi)

// Buat serangkaian titik waktu dari 0 hingga waktu\_total

BUAT array waktu\_analitik dari 0 sampai waktu\_total

// Hitung ketinggian di setiap titik waktu menggunakan rumus eksak

UNTUK setiap t\_a DALAM waktu\_analitik:

SET h\_a = tinggi\_awal - 0.5 \* gravitasi \* t\_a^2

TAMBAHKAN h\_a ke array tinggi\_analitik

AKHIR UNTUK

// -- B. Perhitungan Numerik (Metode Euler) --

DEKLARASIKAN array: waktu\_numerik, tinggi\_numerik

// Kondisi awal

SET t = 0

SET h\_numerik = tinggi\_awal

SET v\_numerik = 0 // Kecepatan awal

// Simpan kondisi awal

TAMBAHKAN t ke waktu\_numerik

TAMBAHKAN h\_numerik ke tinggi\_numerik

// Mulai loop simulasi hingga menyentuh tanah

SELAMA h\_numerik > 0:

// Update kecepatan dan posisi berdasarkan nilai \*sebelumnya\*

SET h\_numerik = h\_numerik - v\_numerik \* dt

SET v\_numerik = v\_numerik + gravitasi \* dt

SET t = t + dt

// Simpan hasil langkah ini

TAMBAHKAN t ke waktu\_numerik

TAMBAHKAN h\_numerik ke tinggi\_numerik

AKHIR SELAMA

// -- C. Perbandingan (Plotting) --

BUAT sebuah plot

PLOT tinggi\_analitik vs waktu\_analitik (sebagai garis solid, "Solusi Analitik")

PLOT tinggi\_numerik vs waktu\_numerik (sebagai penanda/titik, "Solusi Numerik")

TAMBAHKAN Judul: "Perbandingan Solusi Analitik vs Numerik"

TAMBAHKAN Label Sumbu X: "Waktu (detik)"

TAMBAHKAN Label Sumbu Y: "Ketinggian (meter)"

TAMBAHKAN Legenda

TAMPILKAN Plot

SELESAI

1. **Gerak Parabola**

// Bagian 1: Inisialisasi Variabel

BEGIN

// Tentukan konstanta dan parameter awal

SET v0 = 10 // Kecepatan awal dalam m/s

SET theta = 60 // Sudut tembak dalam derajat

SET g = 9.8 // Percepatan gravitasi dalam m/s^2

// Konversi sudut dari derajat ke radian untuk perhitungan

SET theta\_rad = KONVERSI\_KE\_RADIAN(theta)

// Uraikan kecepatan awal menjadi komponen horizontal dan vertikal

SET v0x = v0 \* COS(theta\_rad)

SET v0y = v0 \* SIN(theta\_rad)

// Tampilkan data awal untuk verifikasi

PRINT "Data Awal:"

PRINT " Kecepatan awal (v0) =", v0, "m/s"

PRINT " Sudut elevasi (θ) =", theta, "°"

PRINT " Komponen Kecepatan:"

PRINT " v0x =", v0x, "m/s"

PRINT " v0y =", v0y, "m/s"

END

// Bagian 2: Menghitung Karakteristik Utama Lintasan

BEGIN

// Hitung ketinggian maksimum (H\_max)

SET H\_max = (v0y^2) / (2 \* g)

// Hitung waktu yang diperlukan untuk mencapai ketinggian maksimum

SET t\_max\_height = v0y / g

// Hitung jarak horizontal maksimum (R\_max)

SET R\_max = (v0^2 \* SIN(2 \* theta\_rad)) / g

// Hitung total waktu peluru di udara

SET t\_total = 2 \* t\_max\_height

// Tampilkan hasil perhitungan

PRINT "Ketinggian maksimum (H\_max) =", H\_max, "m"

PRINT "Jarak maksimum (R\_max) =", R\_max, "m"

PRINT "Waktu total di udara =", t\_total, "s"

END

// Bagian 3: Mencari waktu untuk mencapai 2/3 ketinggian maksimum

BEGIN

// Tentukan ketinggian target

SET h\_target = (2/3) \* H\_max

PRINT "Target ketinggian =", h\_target, "m"

// Selesaikan persamaan kuadrat: (g/2)t^2 - v0y\*t + h\_target = 0

SET a = g / 2

SET b = -v0y

SET c = h\_target

// Hitung diskriminan untuk menemukan akar-akar persamaan

SET diskriminan = SQRT(b^2 - 4\*a\*c)

// Hitung dua solusi waktu (saat naik dan saat turun)

SET t\_naik = (-b - diskriminan) / (2\*a)

SET t\_turun = (-b + diskriminan) / (2\*a)

// Tampilkan hasilnya

PRINT "Waktu mencapai 2/3 H\_max (saat naik) =", t\_naik, "s"

PRINT "Waktu mencapai 2/3 H\_max (saat turun) =", t\_turun, "s"

END

// Bagian 4: Mencari waktu untuk mencapai 2/3 jarak maksimum

BEGIN

// Tentukan jarak horizontal target

SET x\_target = (2/3) \* R\_max

PRINT "Target jarak horizontal =", x\_target, "m"

// Hitung waktu dari persamaan gerak horizontal: t = x / v0x

SET t\_2\_3\_range = x\_target / v0x

// Hitung ketinggian (y) pada waktu tersebut

SET y\_at\_2\_3\_range = v0y \* t\_2\_3\_range - 0.5 \* g \* (t\_2\_3\_range^2)

// Tampilkan hasilnya

PRINT "Waktu mencapai 2/3 R\_max =", t\_2\_3\_range, "s"

PRINT "Ketinggian pada saat itu =", y\_at\_2\_3\_range, "m"

END