**Kuis 1 (Tugas\_2)**

**Kelompok Santai**

Github: <https://github.com/JonathanSugijanto/Tugas_2_Santai.git>

Anggota:

* Jonathan Sugijanto (10222007)
* Kevin Winata (10221038)
* Melinda Alberta (10222066)
* Afrah Yani (10222049)

1. **Menggrafik fungsi linear dan kuadratik serta memberi titik potong antarfungsi dan dengan sumbu-x dan y**

Akan dicari titik potong antara kurva dengan menggunakan metode *secant*. Definisikan fungsi sebagai

sehingga dan akan saling berpotongan saat . Rumus iterasi untuk metode *secant* adalah

Karena ada dua titik potong, kita gunakan dua nilai awal, yaitu , , , dan . Berikut adalah program *python* untuk mencari titik potong kedua fungsi tersebut dengan metode *secant*.

import numpy as np

import matplotlib.pyplot as plt

# Definisikan fungsi f(x) dan g(x)

def f(x):

    return -x\*\*2 - 3\*x + 5

def g(x):

    return 2\*x + 4

# Definisikan fungsi h(x) = f(x) - g(x)

# Perpotongan f(x) dan g(x) saat h(x)=0

def h(x):

    return f(x) - g(x)

# Metode Secant

def secant\_method(h, x0, x1, tol=1e-6, max\_iter=100):

    for i in range(max\_iter):

        if abs(h(x1)) < tol:

            return x1

        x2 = x1 - h(x1) \* (x1 - x0) / (h(x1) - h(x0))

        x0, x1 = x1, x2

    return None  # Jika tidak konvergen

# Nilai awal sebagai tebakan lokasi titik potong

x0\_1, x1\_1 = -6, -5   # titik potong pertama

x0\_2, x1\_2 = 0, 1     # titik potong kedua

# Pencarian titik potong

root1 = secant\_method(h, x0\_1, x1\_1)

root2 = secant\_method(h, x0\_2, x1\_2)

if root1 is not None and root2 is not None:

    print(f"Titik potong pertama ditemukan pada x = {root1:.6f}, y = {f(root1):.6f}")

    print(f"Titik potong kedua ditemukan pada x = {root2:.6f}, y = {f(root2):.6f}")

else:

    print("Metode tidak konvergen")

# Plot kurva f(x) dan g(x)

x\_vals = np.linspace(-10, 5, 400)

f\_vals = f(x\_vals)

g\_vals = g(x\_vals)

plt.plot(x\_vals, f\_vals, label='Kurva: y = -x^2 - 3x + 5')

plt.plot(x\_vals, g\_vals, label='Garis: y = 2x + 4')

plt.axhline(0, color='black',linewidth=0.5)

plt.axvline(0, color='black',linewidth=0.5)

# Menampilkan titik potong

if root1 is not None:

    plt.scatter(root1, f(root1), color='red', label=f'Titik potong pertama (x = {root1:.2f})')

if root2 is not None:

    plt.scatter(root2, f(root2), color='blue', label=f'Titik potong kedua (x = {root2:.2f})')

plt.title('Titik Potong antara Kurva dan Garis')

plt.xlabel('x')

plt.ylabel('y')

plt.legend()

plt.grid(True)

plt.show()

import numpy as np

import matplotlib.pyplot as plt

# Definisikan fungsi f(x) dan g(x)

def f(x):

    return x\*\*2 - 9

def g(x):

    return 2\*x + 20

# Metode Secant

def secant\_method(func, x0, x1, tol=1e-5, max\_iter=100):

    for i in range(max\_iter):

        f\_x0 = func(x0)

        f\_x1 = func(x1)

        if abs(f\_x1 - f\_x0) < tol:

            break

        x\_new = x1 - f\_x1 \* (x1 - x0) / (f\_x1 - f\_x0)

        if abs(x\_new - x1) < tol:

            return x\_new

        x0, x1 = x1, x\_new

    return x\_new

# Definisikan fungsi h(x) = f(x) - g(x)

def h(x):

    return f(x) - g(x)

# Nilai tebakan awal untuk titik potong kedua fungsi

intersection\_1 = secant\_method(h, -5, -4)  # Estimasi awal titik potong pertama

intersection\_2 = secant\_method(h, 5, 7)   # Estimasi titik potong kedua

# Titik potong dengan sumbu-y

f\_y\_intercept = f(0)

g\_y\_intercept = g(0)

# Nilai tebakan awal untuk titik potong kedua fungsi dengan sumbu-x

f\_x\_intercept\_1 = secant\_method(f, -4, -2)

f\_x\_intercept\_2 = secant\_method(f, 2, 4)

g\_x\_intercept = secant\_method(g, -12, -9)

# Titik-titik potong

points = [

    (intersection\_1, f(intersection\_1)),    # f(x) dengan g(x)

    (intersection\_2, f(intersection\_2)),    # f(x) dengan g(x)

    (f\_x\_intercept\_1, 0),                   # f(x) dengan sumbu-x

    (f\_x\_intercept\_2, 0),                   # f(x) dengan sumbu-x

    (g\_x\_intercept, 0),                     # g(x) dengan sumbu-x

    (0, f\_y\_intercept),                     # f(x) dengan sumbu-y

    (0, g\_y\_intercept)                      # g(x) dengan sumbu-y

]

# Label untuk titik-titik potong

labels = ['A', 'B', 'C', 'D', 'E', 'F', 'G']

# Plot fungsi f(x) dan g(x)

x\_vals = np.linspace(-10, 10, 400)

f\_vals = f(x\_vals)

g\_vals = g(x\_vals)

plt.figure(figsize=(8,6))

plt.plot(x\_vals, f\_vals, label='f(x) = x^2 - 9', color='blue')

plt.plot(x\_vals, g\_vals, label='g(x) = 2x + 20', color='green')

# Plot titik potong

for i, (x, y) in enumerate(points):

    plt.plot(x, y, 'ro')  # Plotkan titik

    plt.text(x, y, f' {labels[i]} ({x:.2f}, {y:.2f})', fontsize=12, verticalalignment='bottom', horizontalalignment='right')

plt.axhline(0, color='black',linewidth=1)

plt.axvline(0, color='black',linewidth=1)

plt.grid(True)

plt.legend()

plt.title("Grafik f(x) = x^2 - 9 dan g(x) = 2x + 20")

plt.xlabel("x")

plt.ylabel("y")

plt.show()

Berikut hasil *screenshot* kode di *python*.

A computer screen shot of a black screen

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Dan hasil yang ditunjukkan setelah di-*run* adalah

A screen shot of a graph

Description automatically generated

1. Print segitiga sama sisi dan siku-siku di cmd dengan tinggi dan banyak baris yang dapat divariasikan

* Untuk Bintang Rata Kiri

A screenshot of a computer

Description automatically generated

* Untuk Bintang Rata Tengah

A screenshot of a computer

Description automatically generated

1. Tabulasi 100 variabel random (x) pada kolom pertama dan f(x)=sin(X) pada kolom kedua, serta export dalam excel/csv

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Baca data excel/csv dan grafikkan

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated