**LAPORAN PRAKTIKUM**

**MATEMATIKA DISKRIT**



Modul : 2

Alda Syafitri

240306041

Program Studi Teknologi Informasi

Fakultas Dakwah dan Ilmu Komunikasi

Universitas Islam Negeri Mataram

2025

|  |  |
| --- | --- |
| Pertemuan |  |
| Topik |  |
| Repository |  |
| Tanggal |  |

1. Materi praktikum:
2. Relasi dan Fungsi

* Relasi

Relasi adalah hubungan antara anggota dua himpunan.

Jika ada dua himpunan, misalnya:

Himpunan A = {1, 2, 3}

Himpunan B = {a, b, c}

Maka relasi dari A ke B adalah pasangan berurutan (x, y) di mana x ∈ A dan y ∈ B.

Contoh relasi:

R = {(1, a), (2, b), (3, c)}

* Fungsi

Fungsi adalah relasi khusus dari himpunan A ke B, di mana setiap anggota A berpasangan dengan tepat satu anggota B.

Contoh fungsi f dari A ke B: f = {(1, a), (2, b), (3, c)}

Ini disebut fungsi karena setiap elemen di A hanya berpasangan dengan satu elemen di B.

2. Invers Fungsi (Fungsi Invers)

Jika sebuah fungsi f: A → B satu-satu dan onto (bijektif), maka fungsi tersebut memiliki invers, yaitu f⁻¹: B → A.

Sifat fungsi invers:

f(f⁻¹(x)) = x

f⁻¹(f(x)) = x

Contoh:

Misalkan f(x) = 2x + 3

Untuk mencari inversnya:

1. Ubah f(x) menjadi y:

y = 2x + 3

2. Tukar x dan y:

x = 2y + 3

3. Selesaikan untuk y:

y = (x - 3)/2

Jadi, f⁻¹(x) = (x - 3)/2

3. Komposisi Fungsi

Komposisi fungsi adalah penggabungan dua fungsi: jika f: A → B dan g: B → C, maka komposisi g ∘ f didefinisikan sebagai:

(g ∘ f)(x) = g(f(x))

Contoh:

Jika f(x) = x + 2 dan g(x) = 3x, maka:

(g ∘ f)(x) = g(f(x)) = g(x + 2) = 3(x + 2) = 3×+6

4.library pendukung implementasi pada python

A. Math :

- Library math dalam Python menyediakan fungsi-fungsi matematika dasar seperti sin, cos, tan, log, exp, dan lainnya.

- Digunakan untuk operasi matematika yang lebih umum dan tidak memerlukan manipulasi array atau vektor.

- Contoh penggunaan:

Python

import math

angle = math.pi / 4

sine\_value = math.sin(angle)

- Numpy:

- Library numpy menyediakan array yang sangatibel dan berbagai fungsi untuk melakukan operasi pada array tersebut.

- Sangat berguna untuk komputasi numerik yang melibatkan array besar.

- Contoh penggunaan:

python

import numpy as np

angle = np.pi / 4

sine\_values = np.sin(angle)

Kapan Menggunakan Math dan Kapan Menggunakan Numpy Khususnya pada Kasus Trigonometri:

- Gunakan math ketika Anda bekerja dengan sudut dalam radian dan membutuhkan fungsi trigonometri dasar.

- Gunakan numpy ketika Anda bekerja dengan array besar atau memerlukan operasi pada sejumlah besar titik data.

B.Numpy

Perbedaan utama antara math dan numpy adalah:

- math dirancang untuk operasi matematika pada nilai tunggal, sedangkan numpy dirancang untuk operasi matematika pada array dan matriks.

- numpy menyediakan fungsi-fungsi yang lebih canggih dan fleksibel untuk melakukan operasi matematika pada data yang besar dan kompleks.

* Numpy untuk Menyusun domain fungsi

- np.linspace:

-Menghasilkan array dengan nilai yang berada dalam rentang tertentu dengan jumlah elemen yang ditentukan.

-Berguna untuk membuat domain yang merata.

-Contoh:

Python

x = np.linspace(0, 10, 5

print(x) # Output: [0. 2.5 5. ]

- np.arange:

- Menghasilkan array dengan nilai yang berada dalam rentang tertentu.

- Nilai yang dihasilkan adalah bilangan bulat.

C. Python

x = np.arange(0, 10, 2)

print(x) # Output: [0 2 4 6 8]

* perbedaan antara np.linspace dan np.arange:

- np.linspace menghasilkan nilai yang merata dalam rentang, sedangkan np.arange menghasilkan bilangan bulat dalam rentang.

- np.linspace memungkinkan untuk menentukan jumlah elemen, sedangkan np.arange secara otomatis menghasilkan elemen berdasarkan langkahentukan.

c) Matplotlib

- Matplotlib:

- Library untuk membuat grafik dan visualisasi data dalam Python.

- Memungkinkan untuk membuat berbagai jenis plot seperti plot garis, plot titik, plot batang, dan lainnya.

- Contoh penggunaan:

python

import matplotlib.pyplot as plt

x = np.linspace(0, 10, 100)

y = np.sin(x)

plt.plot(x, y)

plt.show()

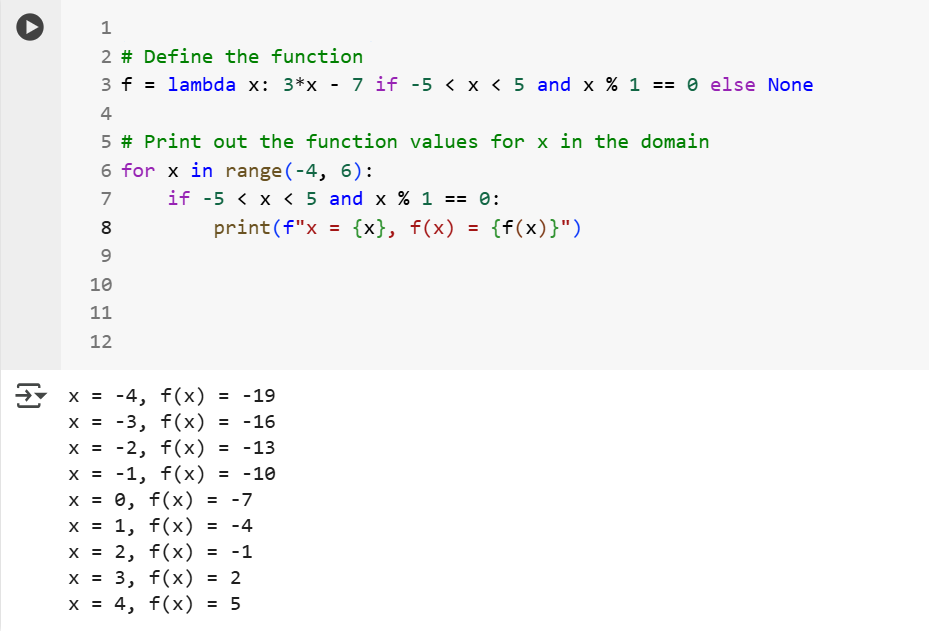
1. Requirement
2. Sistem Operasi yang digunakan : Windows 11 Home Single Language
3. Bowser : Version 129.0.6668.101 (Official

Build) (64-bit)

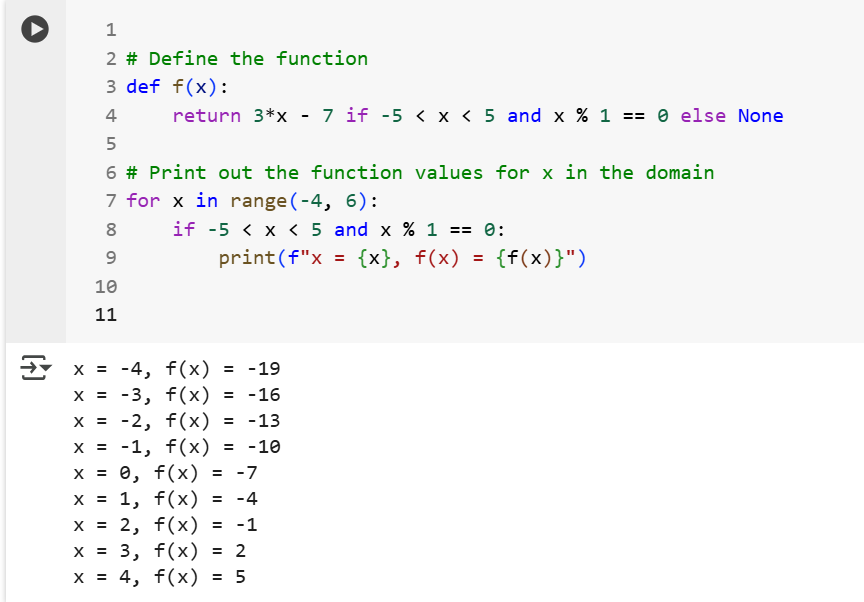
1. Tools yang digunakan : Google colab
2. Implementasi

1. Metode penulisan fungsi pada python

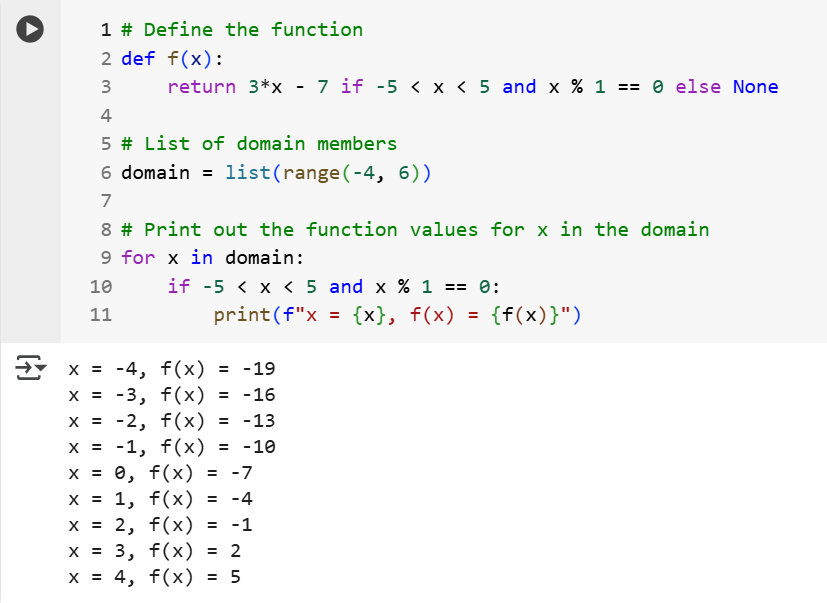
a). Metode konvesional



b.) Dengan menggunakan def



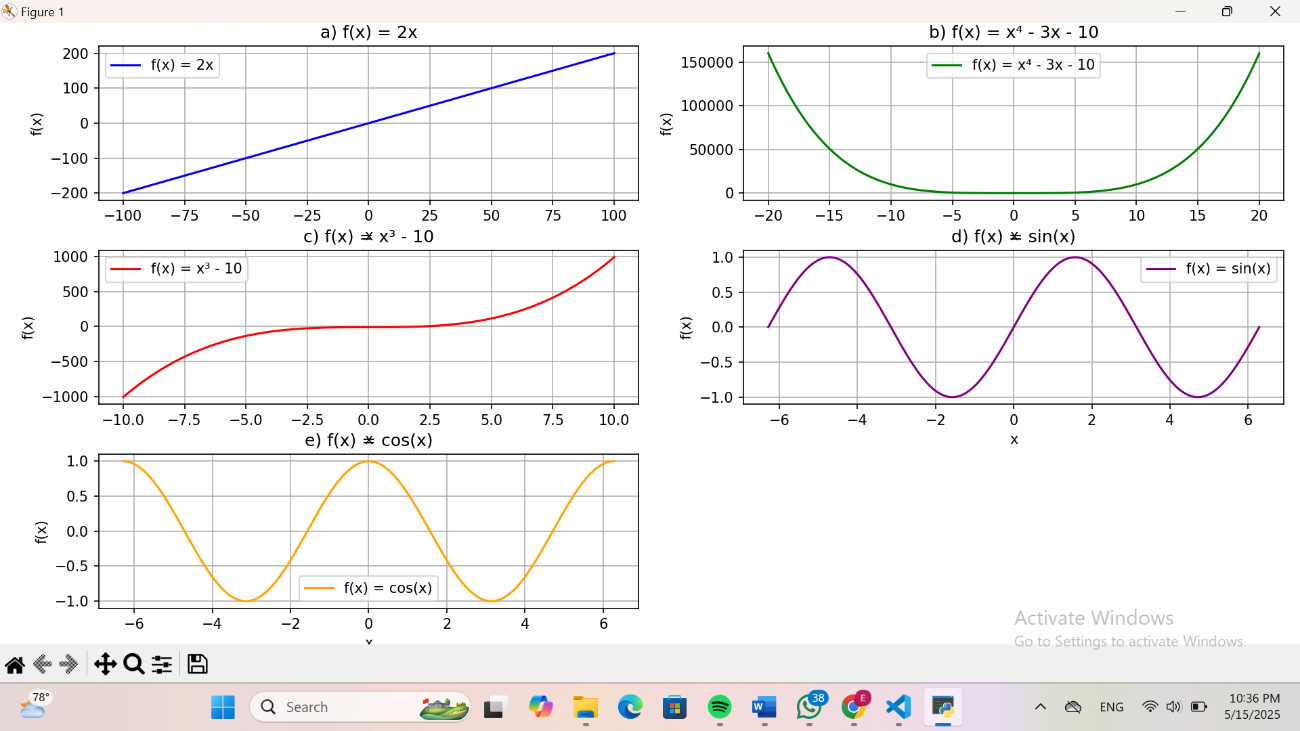
c). Menuliskan anggota himpunan di domain dengan menggunakan list



2). Mendefinisikan fungsi dan menggambarkan grafik fungsi

D. Source Code

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  # a) f(x) = 2x, -100 < x < 100  x\_a = np.linspace(-100, 100, 400)  y\_a = 2 \* x\_a  # b) f(x) = x^4 - 3x - 10, -20 < x < 20  x\_b = np.linspace(-20, 20, 400)  y\_b = x\_b\*\*4 - 3 \* x\_b – 10  # c) f(x) = x^3 - 10, -10 < x < 10  x\_c = np.linspace(-10, 10, 400)  y\_c = x\_c\*\*3 – 10  # d) f(x) = sin(x), -2π < x < 2π  x\_d = np.linspace(-2 \* np.pi, 2 \* np.pi, 400)  y\_d = np.sin(x\_d)  # e) f(x) = cos(x), -2π < x < 2π  x\_e = np.linspace(-2 \* np.pi, 2 \* np.pi, 400)  y\_e = np.cos(x\_e)  # Membuat plot  fig, axs = plt.subplots(3, 2, figsize=(12, 10))  axs = axs.flatten()  # Plot a)  axs[0].plot(x\_a, y\_a, label='f(x) = 2x', color='blue')  axs[0].set\_title('a) f(x) = 2x')  axs[0].set\_xlabel('x')  axs[0].set\_ylabel('f(x)')  axs[0].legend()  axs[0].grid(True)  # Plot b)  axs[1].plot(x\_b, y\_b, label='f(x) = x⁴ - 3x - 10', color='green')  axs[1].set\_title('b) f(x) = x⁴ - 3x - 10')  axs[1].set\_xlabel('x')  axs[1].set\_ylabel('f(x)')  axs[1].legend()  axs[1].grid(True)  # Plot c)  axs[2].plot(x\_c, y\_c, label='f(x) = x³ - 10', color='red')  axs[2].set\_title('c) f(x) = x³ - 10')  axs[2].set\_xlabel('x')  axs[2].set\_ylabel('f(x)')  axs[2].legend()  axs[2].grid(True)  # Plot d)  axs[3].plot(x\_d, y\_d, label='f(x) = sin(x)', color='purple')  axs[3].set\_title('d) f(x) = sin(x)')  axs[3].set\_xlabel('x')  axs[3].set\_ylabel('f(x)')  axs[3].legend()  axs[3].grid(True)  # Plot e)  axs[4].plot(x\_e, y\_e, label='f(x) = cos(x)', color='orange')  axs[4].set\_title('e) f(x) = cos(x)')  axs[4].set\_xlabel('x')  axs[4].set\_ylabel('f(x)')  axs[4].legend()  axs[4].grid(True)  # Sembunyikan subplot ke-6 karena tidak digunakan  fig.delaxes(axs[5]  # Atur layout  plt.tight\_layout()  plt.show() |

1. Output Yang Di Hasilkan