**LAPORAN SIMULASI**

**ALJABAR LINEAR**



Modul 2

Ajewinata

240306009

Program Studi Teknologi Informasi

Fakultas Dakwah dan Ilmu Komunikasi

Universitas Islam Negeri Mataram

2025

|  |  |
| --- | --- |
| Pertemuan | 2 |
| Topik | Operasi Dasar Matriks |
| Repository |  |
| Tanggal | 8 April 2025 |

1. Tujuan

Simulasi ini dilakukan dengan tujuan:

1. Mengenal dan memahami operasi dasar matriks, seperti penjumlahan, pengurangan, perkalian, dan transpose.
2. Menerapkan operasi dasar matriks menggunakan library Numpy.
3. Requirement
4. Sistem Operasi yang digunakan : Windows 11 Version 23H2
5. Tools yang digunakan : Visual Studio Code
6. Durasi Pertemuan : 50 Menit
7. Bahan Simulasi

Hitunglah hasil penjumlahan, pengurangan, perkalian dot, perkalian cross, dan bentuk transpose kedua matriks berikut!

P =[11,12,13]

[14,15,16]

[17,18,19]

Q =[20,21,22]

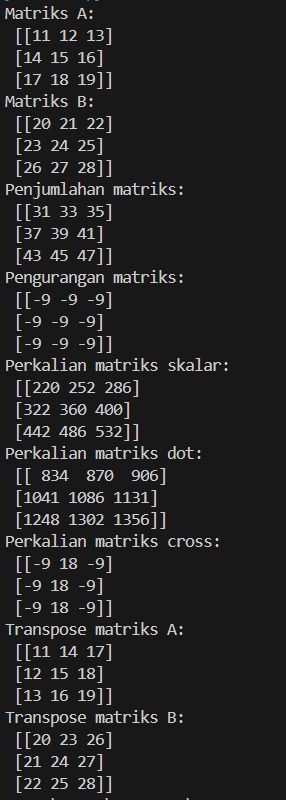
[23,24,25]

[26,27,28]

1. Source Code

|  |
| --- |
| import numpy as np  matriks\_a = np.array([[11,12,13],                  [14,15,16],                        [17,18,19]])  matriks\_b = np.array([[20,21,22],                  [23,24,25],                  [26,27,28]])  penjumlahan\_matriks = matriks\_a + matriks\_b  pengurangan\_matriks = matriks\_a - matriks\_b  perkalian\_skalar = matriks\_a \* matriks\_b  perkalian\_dot = np.dot(matriks\_a,matriks\_b)  perkalian\_cross = np.cross(matriks\_a,matriks\_b)  Transpose\_A = np.transpose(matriks\_a)  Transpose\_B= np.transpose(matriks\_b)  print ("penjumlahan matriks: \n",penjumlahan\_matriks)  print ("pengurangan matriks: \n",pengurangan\_matriks)  print ("Perkalian matriks skalar: \n",perkalian\_skalar)  print ("Perkalian matriks dot: \n",perkalian\_dot)  print ("Perkalian matriks cross: \n",perkalian\_cross)  print ("transpose matriks A: \n",Transpose\_A)  print ("transpose matriks B: \n",Transpose\_B) |

1. Output



1. Penjelasan

|  |
| --- |
| import numpy as np |

Mengimpor library NumPy dengan alias np. NumPy sangat kuat buat operasi matematika, khususnya array dan matriks.

|  |
| --- |
| matriks\_a = np.array([[11,12,13],  [14,15,16],  [17,18,19]]) |

Membuat matriks A ukuran 3x3, berisi angka 11 sampai 19.

|  |
| --- |
| matriks\_b = np.array([[20,21,22],  [23,24,25],  [26,27,28]]) |

Membuat matriks B ukuran 3x3, berisi angka 20 sampai 28.

|  |
| --- |
| penjumlahan\_matriks = matriks\_a + matriks\_b |

Menjumlahkan elemen-elemen yang sejajar dari matriks A dan B.

Contoh: 11 + 20 = 31, 12 + 21 = 33, dst.

|  |
| --- |
| pengurangan\_matriks = matriks\_a - matriks\_b |

Mengurangkan elemen yang sejajar juga.

Contoh: 11 - 20 = -9, 12 - 21 = -9, dst.

|  |
| --- |
| perkalian\_skalar = matriks\_a \* matriks\_b |

Ini disebut perkalian elemen-wise (Hadamard Product), artinya elemen dikali sesuai posisinya.

|  |
| --- |
| perkalian\_dot = np.dot(matriks\_a, matriks\_b) |

Ini adalah perkalian matriks standar (dot product).

Rumusnya: baris di matriks\_a dikalikan dengan kolom di matriks\_b, lalu dijumlahkan.\

|  |
| --- |
| perkalian\_cross = np.cross(matriks\_a, matriks\_b) |

Ini melakukan cross product per baris antar dua matriks (khusus untuk vektor 3D).

Setiap baris matriks\_a[i] dicross dengan matriks\_b[i].

|  |
| --- |
| Transpose\_A = np.transpose(matriks\_a)  Transpose\_B = np.transpose(matriks\_b) |

Fungsi np.transpose() menukar baris jadi kolom, dan sebaliknya.

|  |
| --- |
| print("penjumlahan matriks: \n", penjumlahan\_matriks) |

Menampilkan hasil penjumlahan ke layar.

|  |
| --- |
| print("pengurangan matriks: \n", pengurangan\_matriks) |

Menampilkan hasil pengurangan dari matriks\_a - matriks\_b.

|  |
| --- |
| print("Perkalian matriks skalar: \n", perkalian\_skalar) |

ini mencetak hasil **perkalian elemen satu per satu** (bukan perkalian matriks biasa, tapi **Hadamard Product**).

|  |
| --- |
| print("Perkalian matriks dot: \n", perkalian\_dot) |

Ini mencetak hasil **perkalian dot product antar matriks** (perkalian matriks biasa).

|  |
| --- |
| print("Perkalian matriks cross: \n", perkalian\_cross) |

Ini mencetak hasil cross product antar baris vektor dari matriks\_a dan matriks\_b.

Output-nya juga matriks 3x3, tapi tiap baris adalah hasil np.cross dari baris yang sama dari A dan B.

|  |
| --- |
| print("transpose matriks A: \n", Transpose\_A) |

Menampilkan hasil **transpose** dari matriks\_a. Baris jadi kolom, kolom jadi baris.

|  |
| --- |
| print("transpose matriks B: \n", Transpose\_B) |

Menampilkan hasil **transpose** dari matriks\_b.

1. Kesimpulan

### **1. Komponen Matriks**

**Elemen**  
Matriks terdiri dari elemen-elemen (angka atau simbol) yang disusun dalam struktur baris dan kolom.

**Baris**  
Baris adalah susunan elemen secara horizontal, dari kiri ke kanan.

**Kolom**  
Kolom adalah susunan elemen secara vertikal, dari atas ke bawah.

### **2.Operasi Dasar pada Matriks**

**Penjumlahan Matriks**  
Dua matriks dapat dijumlahkan jika memiliki ukuran (jumlah baris dan kolom) yang sama. Penjumlahan dilakukan dengan menjumlahkan elemen yang bersesuaian.

**Pengurangan Matriks**  
Sama seperti penjumlahan, pengurangan dilakukan antar elemen pada posisi yang sama, dan hanya bisa dilakukan jika ukuran matriks sama.

**Perkalian Matriks**  
Matriks dapat dikalikan dengan matriks lain jika jumlah kolom pada matriks pertama sama dengan jumlah baris pada matriks kedua.

### **3.Aplikasi Matriks dalam Kehidupan Nyata**

**Aljabar Linier**  
Matriks digunakan untuk menyusun dan menyelesaikan sistem persamaan linear, serta untuk transformasi geometri.

**Analisis Data**  
Matriks merepresentasikan kumpulan data dalam bentuk tabel dan digunakan dalam teknik seperti regresi linier dan analisis multivariat.

**Ilmu Komputer**  
Dalam pemrograman dan kecerdasan buatan, matriks digunakan untuk pengolahan citra, graf, machine learning, dan lain-lain.