**LAPORAN SIMULASI**

**ALJABAR LINEAR**



Ajewinata

240306009

Program Studi Teknologi Informasi

Fakultas Dakwah dan Ilmu Komunikasi

Universitas Islam Negeri Mataram

2025

|  |  |
| --- | --- |
| Pertemuan | 1 |
| Topik | Membuat Matriks di Python dan Indeksasi Elemennya |
| Repository | - |
| Tanggal | 20 Maret 2025 |

1. **Tujuan:**

Simulasi ini dilakukan dengan tujuan

1. Memahami penggunaan list untuk merepresentasikan matriks
2. Mengenal dan memahami penggunaan array NumPy untuk membuat matriks
3. Mengenal dan memahami perbedaan list dan array
4. Memahami indeksasi matriks
5. **Requirement**
6. Sistem Operasi yang digunakan : Windows 11 Pro-64 bit

(10.0, Build 26100)

1. Bowser : Version 134.0.6998.36 (Official

Build) (64-bit)

1. Tools yang digunakan : Visual Studio Code
2. **Kompetensi yang diharapkan:**

Setelah mengikuti praktikum ini, peserta diharapkan mampu:

1. Membangun matriks menggunakan list dan array NumPy
2. Mengakses elemen-elemen dalam matriks tersebut
3. Menjelaskan perbedaan list dan array
4. **Durasi Pertemuan: 50 menit**
5. **Bahan Simulasi:**

Tampilkan matriks berikut menggunakan list dan array kemudian indeksasi setiap elemennya!

A =

1. **Hasil Percobaan**
2. Source code

|  |
| --- |
| #menggunakan list  a = [  [-2, 1, -1],  [-3, -1, 2],  [-2, 1, 2]  ]  for i in range(len(a)):  for j in range(len(a[i])):  print(f"a[{i}][{j}] = {a[i][j]}")  #menggunakan array  import numpy as np  a = np.array([  [-2, 1, -1],  [-3, -1, 2],  [-2, 1, 2]  ])  for i in range(a.shape[0]):  for j in range(a.shape[1]):  print(f"a[{i},{j}] = {a[i, j]}") |

1. Hasil Program

|  |
| --- |
|  |

1.menggunakan list

a = [[-2, 1, -1], [-3, -1, 2], [-2, 1, 2]]

Baris ini membuat sebuah array 2 dimensi (dalam Python disebut list of lists).  
Artinya, kamu bikin sebuah matriks berukuran 3 baris dan 3 kolom.  
Setiap elemen bisa diakses dengan indeks baris dan kolom, misalnya a[0][1] artinya baris ke-0, kolom ke-1 (isinya 1).

for i in range(len(a)):

Baris ini memulai perulangan luar (looping baris).  
len(a) akan menghitung jumlah baris dalam array a, yaitu 3, karena ada 3 list di dalamnya.  
Jadi i akan bernilai 0, 1, dan 2 — mewakili baris ke-0, baris ke-1, dan baris ke-2.

for j in range(len(a[i])):

Baris ini memulai perulangan dalam (looping kolom).  
a[i] berarti kita mengambil baris ke-i, dan len(a[i]) menghitung jumlah kolom dalam baris itu (dalam kasus ini selalu 3).  
Jadi j juga akan bernilai 0, 1, dan 2 — mewakili kolom ke-0, kolom ke-1, dan kolom ke-2 di baris itu.

. print(f"a[{i}][{j}] = {a[i][j]}")

Baris ini mencetak nilai elemen array beserta posisi indeksnya.  
Contohnya: kalau i = 1 dan j = 2, maka akan ditampilkan a[1][2] = 2.  
Ini sangat berguna untuk melihat isi dari array 2 dimensi satu per satu dengan urutan yang rapi.

2.Menggunakan array

a = np.array([

[-2, 1, -1],

[-3, -1, 2],

[-2, 1, 2]

])

membuat sebuah array 2 dimensi   
Array a punya 3 baris dan 3 kolom, isinya angka-angka.

for i in range(a.shape[0]):

Baris ini memulai perulangan luar untuk menelusuri baris-baris array.  
a.shape[0] artinya jumlah baris dalam array a, yaitu 3.  
Maka i akan bernilai 0, 1, dan 2 → mewakili baris pertama, kedua, dan ketiga.

for j in range(a.shape[1]):

Ini perulangan di dalam yang digunakan untuk menelusuri kolom dari setiap baris.  
a.shape[1] artinya jumlah kolom dalam array a, yaitu juga 3.  
Jadi j juga akan bernilai 0, 1, dan 2 → mewakili kolom ke-1, 2, dan 3.

print(f"a[{i},{j}] = {a[i, j]}")

Ini adalah perintah untuk menampilkan isi dari array berdasarkan posisi baris dan kolomnya.  
Misalnya:

Saat i = 0 dan j = 0, dicetak a[0,0] = -2

Saat i = 1 dan j = 2, dicetak a[1,2] =

**H. Kesimpulan**

Berdasarkan praktikum ini, dapat disimpulkan bahwa penggunaan NumPy jauh lebih optimal untuk melakukan operasi matriks dibandingkan dengan list biasa. NumPy menawarkan efisiensi yang lebih tinggi, kecepatan proses yang lebih baik, serta dilengkapi dengan berbagai fitur bawaan yang mendukung pengolahan data numerik. Meskipun demikian, list tetap memiliki peran penting sebagai struktur data umum dalam Python, terutama untuk data yang bersifat campuran. Selain itu, pemahaman tentang indeksasi sangat penting agar dapat mengakses dan memanipulasi elemen-elemen dalam matriks atau array dengan tepat dan efektif.

**DAFTAR PUSTAKA**

*https://drive.google.com/file/d/1YAQCbn2yYtZaQ18ypmFsQYhaWMIvqUed/view?usp=drivesdk*