

LAPORAN TUGAS BESAR

MATA KULIAH ALJABAR GEOMETRI

PEMROGRAMAN PERHITUNGAN MATRIKS

MENGGUNAKAN PYTHON



Aldy Abdu Aziz Muzadilah : 10222198
Muhamad Akmal Husnayain : 10222184
Muhamad Fisal Abdul Gani : 10222066
Karlina : 10222092

PROGRAM STUDI INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI CIPASUNG
TASIKMALAYA

2023

BAB I DESKRIPSI MASALAH

Aljabar Geometri merupakan salah satu mata kuliah yang ditempuh di semester tiga program studi informatika Sekolah Tinggi Teknologi Cipasung. Aljabar geometri adalah salah satu cabang ilmu matematika yang sangat penting untuk dipelajari bagi mahasiswa informatika agar memiliki dasar dalam memahami konsep matematika lanjutan dan aplikasinya dalam teknologi informasi.

Dalam mata kuliah ini mahasiswa semester tiga belajar materi matriks. Operasi matriks memiliki peran penting dalam berbagai bidang seperti matematika, ilmu komputer, statistic, dan ilmu lainnya. Dalam konteks ilmu matematika dan ilmu komputer, matriks digunakan untuk mempresentasikan data, memodelkan sistem linier, dan mendukung berbagai operasi numerik.

Untuk tugas besar mahasiswa diharuskan membuat program perhitungan matriks. Matriks yang akan dihitung adalah Penjumlahan dan pengurangan matriks, operasi transpose matriks, operasi determinan matriks, operasi matriks balikan dan SPL (Sistem Persamaan Linier).

Dalam SPL terdapat dua metode yang dipelajari yaitu Metode Eliminasi Gaus dan Metode Eliminasi Gaus-Jordan, determinan matriks adalah perkalian elemen-elemen pada diagonal utama sebuah matriks matriks balikan merupakan matriks yang jika dikalikan dengan matriks aslinya akan menghasilkan matriks identitas, matriks transpose adalah matriks yang diperoleh dengan cara menukar baris menjadi kolom dan kolom menjadi baris, penjumlahan dan penjumlahan matriks adalah menjumlahkan elemen-elemen dari dua buah matriks dengan ordo atau ukuran yang sama.

Untuk membuat program perhitungan matriks ini menggunakan Bahasa pemrograman python dan NumPy untuk operasi matriks. Numpy adalah modul yang sangat populer di kalangan pengembang Phyton untuk komputansi matriks dan manipulasi matriks.

Dengan menggunakan Phyton bersama dengan modul NumPy yang kaya fitur, program ini memberikan kemudahan bagi pengguna dalam menjelajahi operasi matriks dan menerapkan konsep matematika dasar pada data.

BAB II TEORI

2.1 Penjumlahan Matriks dan Pengurangan Matriks

a. Penjumlahan Matriks

Operasi matriks yang pertama kita bahas adalah penjumlahan matriks. Operasi matriks adalah penjumlahan 2 matriks yang letak komponennya sama. Sebenarnya, operasi matriks tidak jauh beda dengan operasi penjumlahan pada umumnya, tapi ada syarat matriks bisa dijumlahkan.

$$\begin{aligned} A + B &= \begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} e & f \\ g & h \end{bmatrix} \\ &= \begin{bmatrix} a + e & b + f \\ c + g & d + h \end{bmatrix} \end{aligned}$$

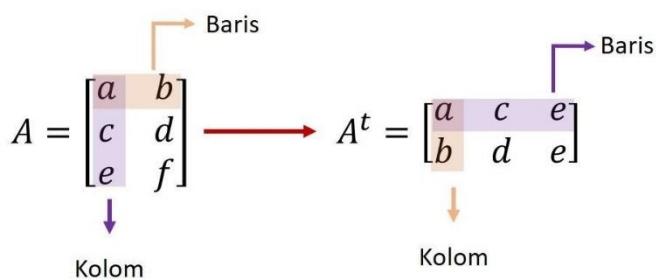
b. Pengurangan Matriks

Syarat pengurangan matriks pun sama dengan penjumlahan, yaitu kedua matriks harus memiliki ordo yang sama sehingga bisa dioperasikan.

$$\begin{aligned} A - B &= \begin{bmatrix} a & b \\ c & d \end{bmatrix} - \begin{bmatrix} e & f \\ g & h \end{bmatrix} \\ &= \begin{bmatrix} a - e & b - f \\ c - g & d - h \end{bmatrix} \end{aligned}$$

2.2 Matriks Transpose

Transpose matriks adalah matriks baru yang elemen baris dan kolomnya merupakan elemen kolom dan baris matriks sebelumnya. Artinya, transpose matriks dibentuk oleh pembalikan elemen baris menjadi kolom dan elemen kolom menjadi baris. Jika matriks yang akan dijadikan transpose bukan matriks persegi, maka ordo pada transposenya merupakan kebalikan dari ordo matriks sebelumnya.



Sifat-sifat matriks transpose :

- a) $(A^t)^t = A$
- b) $(A+B)^t = A^t + B^t$
- c) $(A-B)^t = A^t - B^t$
- d) $(aA)^t = aA^t$
- e) $(AB)^t = B^t A^t$

2.3 Matriks Balikan

Matriks B dikatakan invers dari matriks A, jika $AB=BA=1$ (identitas).

$$A = \begin{bmatrix} \mathbf{a} & \mathbf{b} \\ \mathbf{c} & \mathbf{d} \end{bmatrix} \rightarrow A^{-1} = \frac{1}{ad-bc} \begin{bmatrix} \mathbf{d} & -\mathbf{b} \\ -\mathbf{c} & \mathbf{a} \end{bmatrix}$$

2.4 Determinan Matriks

Determinan Misalkan A adalah matriks berukuran nxn

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{1n} \\ a_{21} & a_{22} & a_{2n} \\ \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{nn} \end{bmatrix}$$

Determinan matriks A dilambangkan dengan :

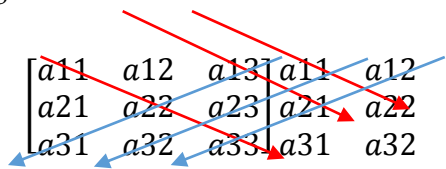
$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & a_{1n} \\ a_{21} & a_{22} & a_{2n} \\ a_{n1} & a_{n2} & a_{nn} \end{vmatrix}$$

1. Determinan matriks 2x2

Determinan matriks A ditulis $\det(A)$ dengan ordo 2x2 diperoleh dengan cara mengurangkan hasil kali elemen-elemen diagonal utama dengan hasil kali elemen-elemen diagonal kedua.

$$A = \begin{bmatrix} \mathbf{a} & \mathbf{b} \\ \mathbf{c} & \mathbf{d} \end{bmatrix} \rightarrow \det(A) = |A| = ad - bc$$

2. Determinan Matriks 3x3

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$


Maka $\det(A) = (a_{11}a_{22}a_{33} + a_{12}a_{21}a_{31} + a_{13}a_{21}a_{32}) - (a_{13}a_{22}a_{31} + a_{11}a_{23}a_{32} + a_{12}a_{21}a_{33})$

2.5 Sistem Persamaan Linier

2.5.1 Metode Eliminasi Gauss

Metode eliminasi gauss adalah metode untuk operasi nilai-nilai dalam matriks, untuk membuat matriks lebih sederhana lagi. Metode eliminasi gauss dikembangkan dari metode eliminasi, dengan cara menghilangkan atau mengurangi jumlah variabel, untuk mendapatkan nilai variabel bebas. Eliminasi Gauss mengubah persamaan linear menjadi bentuk matriks, kemudian diubah ke bentuk Eselon Baris melalui Operasi Baris Elementer. Setelah itu bentuk matriks diselesaikan dengan substitusi balik.

Langkah pengerjaan :

1. Nyatakan SPL dalam bentuk matriks augmented

2. Terapkan OBE pada matriks augmented samapi terbentuk matriks eselon baris
3. Pecahkan persamaan yang berkoresponden pada matriks eselon baris dengan teknik penyulihan mundur (*backward substitution*)

2.5.2 Metode Eliminasi Gaus-Jordan

Metode eliminasi Gaus-Jordan merupakan pengembangan dari metode eliminasi Gaus. Metode ini adalah metode pemecahan suatu sistem persamaan linear dengan mengubahnya menjadi suatu bentuk matriks eselon baris tereduksi dengan metode operasi baris elementer. Tidak diperlukan lagi substitusi secara mundur untuk memperoleh nilai-nilai variable. Nilai variable langsung diperoleh dari matriks augmented akhir. Metode eliminasi Gaus-Jordan terdiri dari dua fase :

1. Fase Maju (forward phase) atau fase eliminasi Gaus, pada fase ini akan menghasilkan nilai-nilai 0 dibawah 1 utama. Contoh :

$$\begin{bmatrix} 2 & 3 & -1 & 5 \\ 4 & 4 & -3 & 3 \\ -2 & 3 & -1 & 1 \end{bmatrix} \sim \text{OBE} \sim \begin{bmatrix} 1 & 3/2 & -1/2 & 5/2 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

2. Fase Mundur (backward phase), pada fase ini akan menghasilkan nilai-nilai 0 di atas 1 utama. Contoh :

$$\begin{bmatrix} 1 & 3/2 & -1/2 & 5/2 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix} \sim \text{OBE} \sim \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{bmatrix} \quad \leftarrow \text{Matriks eselon baris tereduksi}$$

Dari matriks augmented terakhir diperoleh $x_1=1$, $x_2=2$, $x_3=3$

BAB III PENJELASAN IMPLEMENTASI PROGRAM

Program ini adalah implementasi kalkulator matriks menggunakan Python dengan menggunakan library NumPy untuk operasi-operasi matriks. Berikut adalah penjelasan mengenai struktur class yang didefinisikan, garis besar program, dan fungsi-fungsi yang ada:

A. Fungsi-fungsi Utama

1. `input_matrix(rows, cols)`: Fungsi ini meminta pengguna memasukkan elemen-elemen matriks jumlah `rows` dan `cols` dan mengembalikan matriks NumPy hasil input pengguna.
2. `display_matrix(matrix)`: Fungsi ini menampilkan matriks ke layar.
3. `add_matrix_2x2(matrix_a, matrix_b)`: Fungsi ini menjumlahkan dua matriks dengan ordo 2x2.
4. `subtract_matrix_2x2(matrix_a, matrix_b)`: Fungsi ini mengurangi dua matriks dengan ordo 2x2.
5. `transpose_matrix_2x2(matrix)`: Fungsi ini mengembalikan transpose dari suatu matriks dengan ordo 2x2.
6. `transpose_matrix_3x3(matrix)`: Fungsi ini mengembalikan transpose dari suatu matriks dengan ordo 3x3.
7. `determinant_2x2(matrix)`: Fungsi ini menghitung determinan matriks 2x2.
8. `determinant_3x3(matrix)`: Fungsi ini menghitung determinan matriks 3x3.
9. `inverse_2x2(matrix)`: Fungsi ini menghitung invers dari matriks 2x2.
10. `inverse_3x3(matrix)`: Fungsi ini menghitung invers dari matriks 3x3.
11. `solve_linear_system_2x2(coefficients, constants)`: Fungsi ini menyelesaikan sistem persamaan linier untuk matriks koefisien 2x2 dan vektor konstanta. Jadi ordo yang dihasilkan adalah 2x3.
12. `main()`: Fungsi utama yang menjalankan program, program berjalan dalam loop tak terbatas yang meminta pengguna memilih operasi apa yang ingin dilakukan., menu utama mencakup penjumlahan/pengurangan matriks, transpose matriks, invers matriks, menghitung determinan, dan menyelesaikan sistem persamaan linier, dan pengguna dapat keluar dari program dengan memilih opsi keluar.

B. Penggunaan NumPy

- Program menggunakan fungsi-fungsi dari NumPy seperti `np.array`, `np.linalg.det`, `np.linalg.inv`, dan `np.linalg.solve` untuk melakukan operasi matriks.
- Ini membuat implementasi lebih efisien dan mudah dibaca.

C. Main Block

- Pada blok terakhir (`if __name__ == "__main__":`), fungsi `main()` dipanggil untuk menjalankan program saat file ini dijalankan langsung.
- Ini adalah konvensi umum dalam pemrograman Python untuk memastikan bahwa kode di dalam blok ini hanya dijalankan jika file dieksekusi langsung, bukan diimpor sebagai modul.

Note 1 : Dalam program yang diberikan, tidak ada definisi kelas (class) dan oleh karena itu, tidak ada atribut dan metode kelas yang didefinisikan. Semua fungsi-fungsi yang terdapat dalam program tersebut adalah fungsi-fungsi biasa (non-method functions). Dalam konteks pemrograman berorientasi objek, atribut dan metode biasanya didefinisikan dalam kelas. Atribut adalah variabel yang menyimpan informasi tentang suatu objek, sedangkan metode adalah fungsi yang terkait dengan objek tersebut. Namun, di dalam program yang diberikan, implementasinya lebih bersifat prosedural dan menggunakan fungsi-fungsi terpisah untuk melakukan operasi-operasi pada matriks.

Tidak ada kelas yang didefinisikan untuk mewakili objek matriks, dan oleh karena itu tidak ada atribut atau metode kelas. Jika Anda ingin membuat struktur yang lebih terorganisir dengan atribut dan metode, Anda bisa membuat kelas untuk matriks dan menggabungkan fungsi-fungsi ini sebagai metode-metode dari kelas tersebut. Namun, perlu diingat bahwa pendekatan yang diberikan dalam program tersebut adalah valid dan sesuai dengan gaya pemrograman Python yang lebih fungsional.

Note 2 : Dalam program yang diberikan, beberapa fungsi menggunakan NumPy untuk melakukan operasi-operasi matriks. Berikut adalah fungsi-fungsi tersebut:

1. `input_matrix (rows, cols)`: Fungsi ini menggunakan NumPy untuk membuat matriks dari input pengguna. Hasilnya dikembalikan sebagai array NumPy.
2. `add_matrix (matrix_a, matrix_b)`: Fungsi ini menggunakan operasi penjumlahan NumPy (+) untuk menjumlahkan dua matriks.
3. `subtract_matrix (matrix_a, matrix_b)`: Fungsi ini menggunakan operasi pengurangan NumPy (-) untuk mengurangkan dua matriks.
4. `transpose_matrix (matrix)`: Fungsi ini menggunakan metode `^T` dari NumPy untuk menghitung transpose dari suatu matriks.
5. `determinant 2x2 (matrix)`: Fungsi ini menggunakan fungsi `np.linalg.det` dari NumPy untuk menghitung determinan matriks 2x2.
6. `determinant 3x3 (matrix)`: Fungsi ini juga menggunakan fungsi `np.linalg.det` dari NumPy untuk menghitung determinan matriks 3x3.
7. `inverse 2x2 (matrix)`: Fungsi ini menggunakan fungsi `np.linalg.inv` dari NumPy untuk menghitung invers dari matriks 2x2.
8. `inverse 3x3 (matrix)`: Fungsi ini juga menggunakan fungsi `np.linalg.inv` dari NumPy untuk menghitung invers dari matriks 3x3.

9. `solve_linear_system_2x2` (coefficients, constants): Fungsi ini menggunakan fungsi ``np.linalg.solve`` dari NumPy untuk menyelesaikan sistem persamaan linier dengan matriks koefisien 2x2 dan vector konstanta.

10. `solve_linear_system_3x3` (coefficients, constants): Fungsi ini juga menggunakan fungsi ``np.linalg.solve`` dari NumPy untuk menyelesaikan sistem persamaan linier dengan matriks koefisien 3x3 dan vektor konstanta.

Penggunaan NumPy mempermudah implementasi operasi-operasi matriks dan membuat kode lebih efisien dan mudah dibaca.

Fungsi yang tidak menggunakan NumPy :

1. `display_matrix(matrix)`: Menampilkan matriks ke layar menggunakan fungsi bawaan Python ``print``.
2. `main ()`: Membuat struktur utama program tanpa menggunakan NumPy.

Jadi, ada beberapa fungsi yang tidak menggunakan NumPy, tetapi sebagian besar fungsi yang terlibat dalam operasi matriks menggunakan NumPy untuk kenyamanan dan efisiensi.

BAB IV PENGUJIAN

Dalam pengujian ini disajikan eksekusi program. Setiap perhitungan matriks akan diberikan contoh soal masing-masing.

Pilihan Menu

Saat di run tampilan yang pertama kali muncul adalah Menu perhitungan matriks dari 1-5 dan yang ke-6 adalah keluar.

Pengguna hanya perlu menginputkan angka dengan menu yang sesuai dengan keinginan.

```
==SELAMAT DATANG DIOPERASI PERHITUNGAN MATRIKS==  
MENU  
.....  
1. Penjumlahan dan Pengurangan Matriks  
2. Matriks Transpose  
3. Matriks Invers  
4. Determinan  
5. Sistem Persamaan Linier  
6. Keluar  
Pilih menu (1-6):
```

4.1 Pengujian untuk Perhitungan Matriks Penjumlahan dan Pengurangan

4.1.2 Penjumlahan

Contoh soal yang diberikan pada uji penjumlahan adalah :

Jumlahkan matriks

A dan matriks B berikut ini :

$$A = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix}$$

$$B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$



```
Pilih menu (1-6): 1  
Sub Menu :  
1. Penjumlahan  
2. Pengurangan  
Pilih sub-menu untuk penjumlahan/pengurangan matriks ordo 2x2 : 1  
Masukkan elemen matriks  
Masukan elemen matriks 1 kolom 1: 2  
Masukan elemen matriks 1 kolom 2: 3  
Masukan elemen matriks 2 kolom 1: 4  
Masukan elemen matriks 2 kolom 2: 5  
Masukkan elemen matriks  
Masukan elemen matriks 1 kolom 1: 5  
Masukan elemen matriks 1 kolom 2: 6  
Masukan elemen matriks 2 kolom 1: 7  
Masukan elemen matriks 2 kolom 2: 8
```

Hasil dari penjumlahan matriks A dan B adalah :

$$A + B = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 7 & 9 \\ 11 & 13 \end{bmatrix}$$



```
=====  
Hasil penjumlahan matriks A dan B:  
Matriks:  
[[ 7  9]  
 [11 13]]
```

Telah melakukan uji coba memberikan contoh soal untuk matriks penjumlahan dan hasilnya sesuai dengan acuan yang diberikan.

4.1.2 Pengurangan Matriks

Contoh soal yang diberikan

pada uji pengurangan matriks adalah :

Tentukan hasil pengurangan
dari matriks A dan B berikut :

$$A = \begin{bmatrix} 2 & 5 \\ 6 & 7 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 5 \\ 7 & 4 \end{bmatrix}$$



```
Pilih menu (1-6): 1
Sub Menu :
1. Penjumlahan
2. Pengurangan
Pilih sub-menu untuk penjumlahan/pengurangan matriks ordo 2x2 : 2
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 2
Masukan elemen matriks 1 kolom 2: 5
Masukan elemen matriks 2 kolom 1: 6
Masukan elemen matriks 2 kolom 2: 7
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 1
Masukan elemen matriks 1 kolom 2: 5
Masukan elemen matriks 2 kolom 1: 7
Masukan elemen matriks 2 kolom 2: 4
```

Hasil dari pengurangan matriks A dan B adalah :

$$A - B \begin{bmatrix} 2 & 5 \\ 6 & 7 \end{bmatrix} - \begin{bmatrix} 1 & 5 \\ 7 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 3 \end{bmatrix}$$



```
Hasil pengurangan matriks A dan B:
Matriks:
[[ 1 0]
 [-1 3]]
```

Pada matriks pengurangan diberikan contoh soal seperti di atas, dan setelah dieksekusi hasilnya sesuai dengan acuan.

4.2 Pengujian untuk Perhitungan Matriks Transpose

a) Ordo 2x2

Tentukan transpose dari matriks berikut :

$$A = \begin{bmatrix} 2 & 3 \\ 2 & 6 \end{bmatrix}$$



Hasilnya adalah :

$$A^T = \begin{bmatrix} 2 & 3 \\ 2 & 6 \end{bmatrix}$$



```
Pilih menu (1-6): 2
1. Ordo 2x2
2. Ordo 3x3
Pilih sub-menu untuk matriks transpose (1/2): 1
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 2
Masukan elemen matriks 1 kolom 2: 3
Masukan elemen matriks 2 kolom 1: 2
Masukan elemen matriks 2 kolom 2: 6
```

```
Hasil transpose matriks:
Matriks:
[[2 2]
 [3 6]]
```

b) Ordo 3x3

Tentukan transpose dari matriks berikut :

$$A = \begin{bmatrix} 2 & 8 & 9 \\ 6 & 5 & 7 \\ 4 & 4 & 4 \end{bmatrix}$$



```
Pilih menu (1-6): 2
1. Ordo 2x2
2. Ordo 3x3
Pilih sub-menu untuk matriks transpose (1/2): 2
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 2
Masukan elemen matriks 1 kolom 2: 8
Masukan elemen matriks 1 kolom 3: 9
Masukan elemen matriks 2 kolom 1: 6
Masukan elemen matriks 2 kolom 2: 5
Masukan elemen matriks 2 kolom 3: 7
Masukan elemen matriks 3 kolom 1: 4
Masukan elemen matriks 3 kolom 2: 4
Masukan elemen matriks 3 kolom 3: 4
```

$$A^T = \begin{bmatrix} 2 & 6 & 4 \\ 8 & 5 & 4 \\ 9 & 7 & 4 \end{bmatrix}$$



```
Hasil transpose matriks:
Matriks:
[[2 6 4]
 [8 5 4]
 [9 7 4]]
```

4.3 Pengujian untuk perhitungan matriks Invers

Selesaikan matriks invers berikut :

$$A = \begin{bmatrix} 3 & 4 \\ 5 & 6 \end{bmatrix}$$



```
Pilih menu (1-6): 3
Menghitung matriks invers ordo 2x2
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 3
Masukan elemen matriks 1 kolom 2: 4
Masukan elemen matriks 2 kolom 1: 5
Masukan elemen matriks 2 kolom 2: 6
```

$$A = \begin{bmatrix} -3 & 2 \\ 2,5 & -1,5 \end{bmatrix}$$



```
Hasil matriks balikan 2x2:
Matriks:
[[-3.  2. ]
 [ 2.5 -1.5]]
```

4.4 Pengujian untuk perhitungan matriks Determinan

a) Ordo 2x2

Selesaikan determinan matriks berikut:

$$A = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix}$$



```
Pilih menu (1-6): 4
Pilih sub_menu
1. Ordo 2x2
2. Ordo 3x3
Pilih sub-menu untuk determinan (1/2): 1
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 2
Masukan elemen matriks 1 kolom 2: 3
Masukan elemen matriks 2 kolom 1: 4
Masukan elemen matriks 2 kolom 2: 5
```

Hasil dari detreminan matriks di atas adalah -2

```
Determinan matriks 2x2: -2.0
```

b) Ordo 3x3

$$A = \begin{bmatrix} 3 & 5 & 6 \\ 4 & 3 & 7 \\ 3 & 2 & 3 \end{bmatrix}$$



```
Pilih menu (1-6): 4
Pilih sub_menu
1. Ordo 2x2
2. Ordo 3x3
Pilih sub-menu untuk determinan (1/2): 2
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 3
Masukan elemen matriks 1 kolom 2: 5
Masukan elemen matriks 1 kolom 3: 6
Masukan elemen matriks 2 kolom 1: 4
Masukan elemen matriks 2 kolom 2: 3
Masukan elemen matriks 2 kolom 3: 7
Masukan elemen matriks 3 kolom 1: 3
Masukan elemen matriks 3 kolom 2: 2
Masukan elemen matriks 3 kolom 3: 3
```

Hasil dari determinan matriks di atas adalah

```
Determinan matriks 3x3: 24.000000000000000
```

4.5 Pengujian untuk solusi Sistem Persamaan Linier (SPL)

Dalam pengujian SPL ini langkah pertama memilih menu, kemudian memasukan elemen baris dengan ordo 2x2. Setelah itu masukan nilai konstantanya.

Selesaikan SPL berikut :

$$2X_1 + 3X_2 = 5$$

$$4X_1 + 5X_2 = 6$$



```
Pilih menu (1-6): 5
Menghitung SPL 2X3
Masukkan elemen matriks
Masukan elemen matriks 1 kolom 1: 2
Masukan elemen matriks 1 kolom 2: 3
Masukan elemen matriks 2 kolom 1: 4
Masukan elemen matriks 2 kolom 2: 5
Masukkan konstanta (baris 1, baris 2): 5 6
```

Hasilnya adalah :



```
Hasil sistem persamaan linier 2x2:
x = -3.5
y = 4.0
```

BAB V KESIMPULAN

5.1 Kesimpulan

Matriks merupakan alat penting dalam berbagai bidang seperti matematika, ilmu komputer, statistika dan ilmu-ilmu lainnya. Operasi matriks berperan penting dalam memodelkan sistem linier, mendukung berbagai operasi numerik, dan menyelesaikan sistem persamaan linier.

Program perhitungan matriks yang dibuat dengan bahasa pemrograman Python dan modul NumPy dapat melakukan berbagai operasi matriks seperti penjumlahan, pengurangan, perkalian, perkalian skalar, determinan matriks dan invers. Program ini dapat menjadi alat yang berguna bagi mahasiswa dan praktisi dalam mempelajari dan menerapkan operasi matriks. Cerminan materi terlampir menunjukkan bahwa siswa mempunyai pemahaman yang baik tentang operasi matriks. Mahasiswa mampu menjelaskan teori operasi matriks secara jelas dan sistematis. Selain itu, siswa mampu menerapkan teori ini pada program kalkulator matriks.

5.2 Saran

Karena keterbatasan ilmu kami, program ini masih perlu dikembangkan lagi. Karena program yang kami buat tidak berbasis GUI, untuk kedepannya kami ingin membuat program yang berbasis GUI agar lebih nyaman untuk dipakai. Dengan menggunakan numpy di python ini memudahkan kami dalam pengerjaan program, karena dalam numpy sudah tersedia rumus matriks.

5.3 Repleksi

- a). Mahasiswa harus mempunyai pemahaman yang baik tentang konsep dasar matriks. Konsep dasar matriks merupakan landasan penting untuk memahami operasi matriks yang lebih kompleks.
- b). Mahasiswa hendaknya berlatih secara teratur untuk meningkatkan keterampilannya dalam operasi matriks. Latihan teratur dapat membantu siswa memahami konsep matriks dan menerapkannya pada situasi yang berbeda.
- C). Mahasiswa perlu menggunakan teknologi untuk membantu mereka mempelajari matriks. Teknologi seperti perangkat lunak matriks dan kalkulator matriks dapat menjadi alat yang berguna untuk mempelajari matriks.

REFERENSI

- Fajri, D. L. (2023, Februari 25). *Memahami Metode Eliminasi Gauss dan Pembahasan Soal*. Retrieved from Katadata.co.id:
<https://katadata.co.id/intan/lifestyle/63f8b44a67a6f/memahami-metode-eliminasi-gauss-dan-pembahasan-soal>
- Munir, R. (n.d.). *Determinan*. Retrieved from Institut Teknologi Bandung:
<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-08-Determinan-bagian1.pdf>
- Munir, R. (n.d.). *Sistem Persamaan Linier*. Retrieved from Institut Teknologi Bnadung:
<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-05-Sistem-Persamaan-Linier-2.pdf>
- Transpose Matriks: Konsep, Contoh Soal, dan Pembahasan*. (2022, Juli 31). Retrieved from IDM TIMES:
[https://www.idntimes.com/science/discovery/laili-zain-damaika-1/tranpose-matriks#:~:text=Transpose%20matriks%20juga%20memiliki%20sifat,B\)T%20%3D%20AT%20%E2%80%93%20BT](https://www.idntimes.com/science/discovery/laili-zain-damaika-1/tranpose-matriks#:~:text=Transpose%20matriks%20juga%20memiliki%20sifat,B)T%20%3D%20AT%20%E2%80%93%20BT)
- Wilman Juniardi & Pamela Natasa, S. (2023, Januari 31). *Pengertian Transpose Matriks, Sifat dan Bentuk Penulisannya*. Retrieved from Quiper :
<https://www.quipper.com/id/blog/mapel/matematika/transpose-matriks/>