

LAPORAN TUGAS BESAR ALJABAR GEOMETRI

Laporan ini dibuat untuk memenuhi tugas besar

Mata Kuliah IF 1313 Aljabar Geometri



Disusun Oleh :

Kelompok 7

Rizki reza : 10222104

Gerry Gestario : 10222112

Ai Siti Hasanah : 10222126

Ai Hamidah : 10222100

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI CIPASUNG
TASIKMALAYA
2023**

KATA PENGANTAR

Segala puji dan syukur penulis panjatkan kepada Allah SWT, atas seluruh curahan rahmat dan hidayahnya sehingga penyusun mampu menyelesaikan salah satu Ujian Akhir Semester Mata Kuliah Aljabar Geometri. Laporan ini ditujukan sebagai salah satu syarat mata kuliah Aljabar Geometri di Program Studi Informatika Sekolah Tinggi Teknologi Cipasung. Dalam penyelesaian laporan ini, penulis banyak memperoleh bantuan baik pengajar, bimbingan dan arahan dari berbagai pihak. Untuk itu penulis menyampaikan terima kasih sebesar-besarnya kepada semua pihak yang telah ikut terlibat.

Menyadari sepenuhnya bahwa dalam penulisan laporan ini masih jauh dari kata sempurna karena keterbatasan kemampuan dan ilmu pengetahuan yang dimiliki penulis, oleh karena itu atas kesalahan dan kekurangan dalam laporan ini penulis bersedia menerima saran dan kritikan yang sifatnya membangun dari pembaca. Semoga Laporan ini dapat memberikan manfaat bagi siapa saja yang membacanya.

Tasikmalaya, Desember 2023

Penulis

DAFTAR ISI

| | |
|--|-----------|
| DAFTAR ISI | ii |
| BAB 1..... | 1 |
| DESKRIPSI MASALAH..... | 1 |
| BAB II..... | 3 |
| LANDASAN TEORI | 3 |
| A. Sistem persamaan linier | 3 |
| B. Determinan | 4 |
| C. Matriks balikan | 5 |
| D. Matriks transpose | 5 |
| BAB III Penjelasan Implementasi Program | 8 |
| BAB IV Pengujian | 19 |
| BAB V Kesimpulan..... | 31 |

BAB 1

DESKRIPSI MASALAH

Setiap kelompok diminta untuk membuat sebuah program untuk menemukan solusi SPL dengan metode eliminasi Gauss, metode Eliminasi Gauss-Jordan, matriks balikan, Matriks Transpose, menghitung determinan matriks, dan menghitung penjumlahan dan pengurangan matriks.

Adapun spesifikasi program adalah sebagai berikut :

1. Program dapat menerima masukan (input) baik dari keyboard maupun membaca masukan dari file text.
2. Untuk persoalan menghitung matriks, masukan dari keyboard adalah dua buah matriks (matriks A dan B) dengan setiap nilai dalam matriksnya (a_{11} , a_{12} , a_{21} , a_{22} , b_{11} , b_{12} , b_{21} dan b_{22})
3. Untuk persoalan SPL, luaran (output) program adalah solusi SPL. Jika solusinya tunggal, tuliskan nilainya. Jika solusinya tidak ada, tuliskan solusi tidak ada, jika solusinya banyak, maka tuliskan solusinya dalam bentuk parametrik (misalnya $x_4 = -2$, $x_3 = 2s - t$, $x_2 = s$, dan $x_1 = t$.)
4. Untuk persoalan determinan dan matriks balikan, maka luarannya sesuai dengan persoalan masing-masing
5. Luaran program harus dapat ditampilkan pada layar komputer dan dapat disimpan ke dalam file.
6. Bahasa program yang digunakan adalah python.
7. Program tidak harus berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas Eclipse misalnya).
8. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya,

MENU

1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan

- 5. Sistem Persamaan Linier
- 6. Keluar

Untuk pilihan menu nomor 1 ada sub-menu lagi yaitu pilihan:

- 1. Penjumlahan matriks
- 2. Pengurangan matriks

Untuk pilihan menu nomor 2 ada sub-menu lagi yaitu pilihan:

- 1. Matriks 2×2
- 2. Matriks 3×3

Untuk pilihan menu nomor 4 ada sub-menu lagi yaitu pilihan:

- 1. Matriks 2×2
- 2. Matriks 3×3

BAB II

LANDASAN TEORI

A. Sistem persamaan linier

Persamaan ini disebut linear lantaran hubungan matematis ini dapat digambarkan sebagai garis lurus dalam sistem koordinat kartesius. Apabila terdapat lebih dari satu persamaan linear, persamaan tersebut akan menjadi sebuah sistem.

Bentuk umum SPL

- Linier: pangkat tertinggi di dalam variabelnya sama dengan 1
- Sebuah SPL dengan m buah persamaan dan n variabel x_1, x_2, \dots, x_n berbentuk:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$$

$$\vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \qquad \qquad \vdots$$

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m$$

atau dalam bentuk $Ax = b$

1. Metode eliminasi Gauss

Metode eliminasi Gauss adalah metode untuk operasi nilai-nilai dalam matriks, untuk membuat matriks lebih sederhana lagi. Metode eliminasi gaus dikembangkan dari metode eliminasi, dengan cara menghilangkan atau mengurangi jumlah variabel, untuk mendapatkan nilai variabel bebas. Metode Eliminasi Gauss dikembangkan oleh Carl Friedrich Gauss (1777-1855). Carl adalah matematikawan berkebangsaan Jerman yang berkontribusi dalam geometri, teori bilangan, fungsi, dan teori probabilitas

- Nyatakan SPL dalam bentuk matriks augmented
- Terapkan OBE pada matriks augmented sampai terbentuk matriks eselon baris
- Pecahkan persamaan yang berkoresponden pada matriks eselon baris dengan teknik penyulihan mundur (backward substitution)

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_n \end{bmatrix} \sim_{\text{OBE}} \begin{bmatrix} 1 & * & * & \dots & * & * \\ 0 & 1 & * & \dots & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \vdots & 1 & * \end{bmatrix}$$

2. Metode eliminasi Gauss-Jordan

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. Metode Eliminasi Gauss dikembangkan oleh Carl Friedrich Gauss (1777-1855). Carl adalah matematikawan berkebangsaan Jerman yang berkontribusi dalam geometri, teori bilangan, fungsi, dan teori probabilitas

Merupakan pengembangan metode eliminasi Gauss

- Operasi baris elementer (OBE) diterapkan pada matriks augmented sehingga menghasilkan matriks eselon baris tereduksi.
- Tidak diperlukan lagi substitusi secara mundur untuk memperoleh nilai variabel. Nilai variabel langsung diperoleh dari matriks augmented akhir.

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_m \end{bmatrix} \sim_{\text{OBE}} \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & * \\ 0 & 1 & 0 & \dots & 0 & * \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \vdots & 1 & * \end{bmatrix}$$

B. Determinan

Determinan matriks merupakan selisih antara perkalian elemen-elemen pada diagonal utama dengan perkalian elemen-elemen pada diagonal sekunder. Determinan matriks hanya dapat dicari dengan matriks persegi. Determinan dari matriks A dapat ditulis $\det(A)$ atau $|A|$. Determinan matriks dapat ditemukan dalam matriks persegi ordo 2x2 dan 3x3.

1. Determinan Matriks Persegi Berordo 2x2

$$\text{Matriks } A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Hasil kali elemen-elemen diagonal utama dikurangi hasil kali elemen-elemen diagonal samping disebut determinan matriks A. Atau dapat dituliskan dengan $\det A = ad - bc$

2. Determinan Matriks Persegi Berordo 3x3

Sama dengan determinan matriks ordo 2x2, dalam mencari determinan matriks A digunakan cara diagonal utama dikurangi hasil kali elemen-elemen diagonal samping. Namun, pada matriks persegi berordo 3x3 memiliki cara yang berbeda. Berikut penjabarannya

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} - a_{12}a_{23}a_{31} - a_{13}a_{21}a_{32}$$

C. Matriks balikan

Sebuah matriks persegi, matriks yang kolom dan barisnya sama, bisa memiliki sebuah invers atau balikan. Matriks balikan atau matriks invers akan menghasilkan matriks identitas ketika dikalikan dengan matriks asalnya. Sebuah matriks tidak akan memiliki inverse jika determinannya bernilai nol. Misalkan ada sebuah matriks A, maka matriks balikannya dilambangkan dengan A^{-1} . Dan sesuai definisi,

$$A \cdot A^{-1} = I$$

D. Matriks transpose

- Transpose matriks, $B = A^T$ bji = aij $i = 1, 2, \dots, m; j = 1, 2, \dots, n$
- Algoritma transpose matriks:


```

      for i ← 1 to m do
        for j ← 1 to n do
          bji ← aij
        end for
      end for
      
```


$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 3 \\ 1 & 4 \\ 5 & 6 \end{bmatrix}, \quad C = [1 \ 3 \ 5], \quad D = [4]$$

$$A^T = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{32} \\ a_{13} & a_{23} & a_{33} \\ a_{14} & a_{24} & a_{34} \end{bmatrix}, \quad B^T = \begin{bmatrix} 2 & 1 & 5 \\ 3 & 4 & 6 \end{bmatrix}, \quad C^T = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix}, \quad D^T = [4]$$

- Untuk matriks persegi A berukuran $n \times n$, transpose matriks A dapat diperoleh dengan mempertukarkan elemen yang simetri dengan diagonal utama:

$$A = \begin{bmatrix} 1 & -2 & 4 \\ 3 & 7 & 0 \\ -5 & 8 & 6 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -2 & 4 \\ 3 & 7 & 0 \\ -5 & 8 & 6 \end{bmatrix} \rightarrow A^T = \begin{bmatrix} 1 & 3 & -5 \\ -2 & 7 & 8 \\ 4 & 0 & 6 \end{bmatrix}$$

- Sifat-Sifat Transpose Matri
 1. $(P^T)^T = P$
 2. $(P + Q)^T = P^T + Q^T$
 3. $(P - Q)^T = P^T - Q^T$
 4. $(kP)^T = k.P^T$, di mana k = suatu konstanta
 5. $(PQ)^T = P^T Q^T$

E. Penjumlahan dan Pengurangan matriks

Dua buah matriks dapat dijumlahkan apabila keduanya memiliki ordo yang sama. Hasil operasi penjumlahannya adalah matriks baru yang memiliki ordo sama dengan matriks semula, dengan elemen-elemennya terdiri dari hasil penjumlahan elemen-elemen pada matriks. Secara matematis, operasi penjumlahan matriks dapat diasumsikan sebagai berikut:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} p & q \\ r & s \end{bmatrix} = \begin{bmatrix} a+p & b+q \\ c+r & d+s \end{bmatrix}$$

Pengurangan matriks memiliki konsep yang sama dengan penjumlahan. Dua buah matriks dapat dikurangkan apabila keduanya memiliki ordo yang sama. Hasil operasi pengurangannya adalah matriks baru yang memiliki ordo sama dengan matriks semula, dengan elemen-elemennya terdiri dari hasil pengurangan dengan elemen-elemen pada matriks. Secara matematis, operasi pengurangan matriks dapat diasumsikan sebagai berikut:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} - \begin{bmatrix} p & q \\ r & s \end{bmatrix} = \begin{bmatrix} a-p & b-q \\ c-r & d-s \end{bmatrix}$$

BAB III**PENJELASAN IMPLEMENTASI PROGRAM**

Berikut ini adalah penjelasan implementasi program

| Program | Deskripsi |
|---------------------------------------|---|
| <code>def print_matrix(matrix)</code> | Mencetak matriks yang diberikan ke konsol atau output. Fungsi ini berguna untuk memvisualisasikan matriks saat bekerja dengan operasi matriks di dalam program. |
| <code>for row in matrix</code> | Untuk mengiterasi melalui setiap baris dalam matriks (matrix). Dalam konteks ini, matriks diasumsikan sebagai struktur data dua dimensi di mana setiap elemen matriks adalah baris. |
| <code>print(row)</code> | Digunakan untuk mencetak setiap baris matriks ke konsol. Dalam konteks loop <code>for row in matrix:</code> , setiap iterasi dari loop akan mengambil satu baris dari matriks, dan pernyataan <code>print(row)</code> akan mencetak baris tersebut ke konsol. |
| <code>def menu()</code> | Digunakan untuk menampilkan pilihan menu kepada pengguna. Fungsi ini membantu membuat antarmuka pengguna yang jelas dan memandu pengguna untuk memilih opsi yang diinginkan. |
| <code>print("MENU")</code> | Digunakan untuk mencetak teks "MENU" ke konsol. Dalam konteks program, ini biasanya digunakan untuk memberikan pengguna informasi tentang jenis menu yang tersedia. Dalam konteks fungsi <code>menu()</code> , ini |

| | |
|--|--|
| | memberikan informasi kepada pengguna tentang pilihan pertama dalam menu. |
| <code>print("1. Penjumlahan dan Pengurangan Matriks")</code> | Digunakan untuk mencetak opsi menu tertentu ke konsol. |
| <code>print("2. Matriks Transpose")</code> | Digunakan untuk mencetak opsi menu tertentu ke konsol. |
| <code>print("3. Matriks Balikan")</code> | Digunakan untuk mencetak opsi menu tertentu ke konsol. |
| <code>print("4. Determinan")</code> | Digunakan untuk mencetak opsi menu tertentu ke konsol. |
| <code>print("5. Sistem Persamaan Linier")</code> | Digunakan untuk mencetak opsi menu tertentu ke konsol. |
| <code>print("6. Keluar")</code> | Digunakan untuk mencetak opsi keluar ke konsol dalam konteks menu. Ini memberikan opsi kepada pengguna untuk keluar dari program atau mengakhiri interaksi dengan program. |
| <code>def matrix_operations_menu()</code> | Untuk menampilkan opsi menu khusus terkait operasi matriks, seperti penjumlahan dan pengurangan matriks. |
| <code>print("PILIHAN")</code> | Digunakan dalam fungsi <code>matrix_operations_menu()</code> untuk mencetak teks "PILIHAN" ke konsol. Ini membantu memberikan informasi visual kepada pengguna bahwa mereka sekarang memilih dari opsi khusus terkait operasi matriks. |
| <code>print("1. Penjumlahan matriks")</code> | Digunakan dalam fungsi <code>matrix_operations_menu()</code> untuk mencetak opsi penjumlahan matriks ke konsol. |

| | |
|--|---|
| <code>print("2. Pengurangan matriks")</code> | Digunakan dalam fungsi <code>matrix_operations_menu()</code> untuk mencetak opsi pengurangan matriks ke konsol. |
| <code>def transpose_menu()</code> | Digunakan untuk menampilkan opsi menu khusus terkait operasi transpose matriks. |
| <code>print("PILIHAN")</code> | Mencetak teks "PILIHAN" ke konsol. |
| <code>print("1. Matriks 2x2")</code> | Mencetak opsi spesifik untuk melakukan operasi transpose pada matriks berordo 2x2 ke konsol. |
| <code>print("2. Matriks 3x3")</code> | Mencetak opsi spesifik untuk melakukan operasi transpose pada matriks berordo 3x3 ke konsol. |
| <code>def determinant_menu()</code> | Digunakan untuk menampilkan opsi menu khusus terkait perhitungan determinan matriks. |
| <code>print("PILIHAN")</code> | Mencetak teks "PILIHAN" ke konsol. |
| <code>print("1. Matriks 2x2")</code> | Mencetak opsi spesifik untuk melakukan operasi transpose pada matriks berordo 2x2 ke konsol. |
| <code>print("2. Matriks 3x3")</code> | Mencetak opsi spesifik untuk melakukan operasi transpose pada matriks berordo 3x3 ke konsol. |
| <code>def matrix_inverse(matrix)</code> | Digunakan untuk menghitung invers matriks. |
| <code>det = np.linalg.det(matrix)</code> | Digunakan untuk menghitung determinan matriks menggunakan fungsi <code>det</code> dari modul NumPy (<code>np</code>). NumPy menyediakan metode yang efisien untuk operasi matriks, termasuk perhitungan determinan. |
| <code>if det == 0</code> | Digunakan untuk memeriksa apakah determinan matriks (<code>det</code>) sama dengan nol. |

| | |
|---|--|
| <code>return None # Matriks tidak memiliki invers karena determinannya nol</code> | Mengembalikan None ketika determinan matriks adalah nol menunjukkan bahwa matriks tersebut tidak memiliki invers. |
| <code>else</code> | Dijalankan jika determinan matriks tidak sama dengan nol. |
| <code>return np.linalg.inv(matrix)</code> | Akan dijalankan jika determinan matriks tidak sama dengan nol. Pernyataan ini mengembalikan invers dari matriks yang diberikan menggunakan fungsi <code>np.linalg.inv()</code> dari modul NumPy. |
| <code>def linear_system_solution_menu()</code> | Fungsi ini mencetak petunjuk kepada pengguna untuk memasukkan koefisien matriks A dan vektor b. Selanjutnya, fungsi memanggil fungsi <code>input_matrix()</code> untuk meminta input matriks dari pengguna. Akhirnya, fungsi mengembalikan matriks A dan vektor b. |
| <code>print("Masukkan koefisien matriks A dan vektor b:")</code> | Mencetak untuk koefisien matriks A dan vektor b. |
| <code>A = input_matrix(2)</code> | Untuk menginputkan matrix. |
| <code>B = input_matrix(2)</code> | Untuk menginputkan matrix. |
| <code>return A, b</code> | Mengembalikan nilai. |
| <code>while True</code> | Menunjukkan bahwa program akan terus berjalan dalam loop tak terbatas. Ini biasanya digunakan untuk membuat program berjalan secara berulang tanpa henti sampai suatu kondisi tertentu dipenuhi. |
| <code>menu()</code> | Untuk menampilkan opsi-opsi menu kepada pengguna. |

| | |
|--|--|
| <code>choice = input("Pilih menu (1-6): ")</code> | Digunakan untuk meminta input dari pengguna berupa nomor menu. Dengan menggunakan fungsi <code>input()</code> , program akan menampilkan pesan "Pilih menu (1-6): " dan menunggu pengguna memasukkan nilai. |
| <code>if choice == '1'</code> | Menunjukkan bahwa program akan menjalankan serangkaian instruksi jika pengguna memilih menu dengan nomor 1. |
| <code>matrix_operations_menu()</code> | Memberikan pilihan yang lebih spesifik kepada pengguna setelah mereka memilih menu utama terkait operasi matriks. Fungsi ini akan mencetak pesan ke layar dengan opsi-opsi yang dapat dipilih oleh pengguna. |
| <code>operation_choice = input("Pilih operasi (1-2): ")</code> | Meminta pengguna memilih jenis operasi matriks, yaitu penjumlahan atau pengurangan. Setelah pengguna memasukkan nilai, nilai tersebut disimpan dalam variabel <code>operation_choice</code> untuk digunakan dalam langkah selanjutnya dalam program. |
| <code>if operation_choice == '1'</code> | Program akan menjalankan serangkaian instruksi jika pengguna memilih operasi penjumlahan matriks. |
| <code>A = input_matrix(2)</code> | Untuk menginputkan nilai matriks. |
| <code>B = input_matrix(2)</code> | Untuk menginputkan nilai matriks. |
| <code>result = A + B</code> | Bertujuan untuk melakukan operasi penjumlahan matriks. |
| <code>if operation_choice == '2'</code> | Program akan menjalankan serangkaian instruksi jika pengguna memilih operasi penjumlahan matriks. |
| <code>A = input_matrix(2)</code> | Untuk menginputkan nilai matriks. |

| | |
|---|---|
| <code>B = input_matrix(2)</code> | Untuk menginputkan nilai matriks. |
| <code>result = A - B</code> | Bertujuan untuk melakukan operasi pengurangan matriks. |
| <code>else</code> | menangani situasi di mana <code>operation_choice</code> tidak sama dengan '1' atau '2'. Dalam konteks ini, ini mencakup pilihan yang tidak valid yang dibuat oleh pengguna saat memilih operasi matriks. |
| <code>print("Pilihan tidak valid")</code> | Untuk mencetak dan memberi tahu pengguna bahwa pilihan yang mereka buat tidak sesuai dengan opsi yang diharapkan. Ini adalah pesan umum yang berguna untuk memberikan umpan balik kepada pengguna dalam kasus input yang tidak valid. |
| <code>print("Hasil:")</code> | Memberikan penanda bahwa hasil operasi atau output dari program akan dicetak setelah pesan tersebut. |
| <code>print_matrix(result)</code> | Mencetak hasil operasi matriks (penjumlahan atau pengurangan) ke layar. |
| <code>elif choice == '2'</code> | Menunjukkan bahwa program akan menjalankan serangkaian instruksi jika pengguna memilih menu dengan nomor 2. |
| <code>transpose_menu()</code> | Fungsi ini mencetak pesan ke layar dengan opsi-opsi yang dapat dipilih oleh pengguna. menampilkan submenu yang berkaitan dengan pilihan operasi matriks transpose. |
| <code>transpose_choice = input("Pilih operasi (1-2):")</code> | Meminta pengguna memilih jenis operasi matriks transpose, yaitu apakah akan diaplikasikan pada matriks 2x2 atau 3x3. Setelah pengguna memasukkan nilai, nilai |

| | |
|---|--|
| | tersebut disimpan dalam variabel <code>transpose_choice</code> untuk digunakan dalam langkah selanjutnya dalam program. |
| <code>if transpose_choice == '1'</code> | Program akan menjalankan serangkaian instruksi jika pengguna memilih operasi transpose untuk matriks 2x2. |
| <code>A = input_matrix(2)</code> | Memasukkan nilai-nilai elemen matriks. |
| <code>result = np.transpose(A)</code> | Digunakan untuk menjalankan operasi transpose pada matriks A. Operasi transpose akan menukar baris dan kolom matriks, menghasilkan matriks baru. Dalam konteks ini, matriks hasil transpose disimpan dalam variabel <code>result</code> . |
| <code>elif transpose_choice == '2'</code> | Program akan menjalankan serangkaian instruksi jika pengguna memilih operasi transpose untuk matriks 3x3. |
| <code>A = input_matrix(3)</code> | Memasukkan nilai-nilai elemen matriks. |
| <code>result = np.transpose(A)</code> | Menjalankan operasi transpose pada matriks A dengan ordo 3x3. Operasi transpose ini menukar baris dan kolom matriks, menghasilkan matriks baru yang disimpan dalam variabel <code>result</code> . |
| <code>else</code> | Untuk menangani situasi di mana kondisi pada pernyataan <code>if</code> atau <code>elif</code> sebelumnya tidak terpenuhi. Ini memberikan blok kode yang akan dieksekusi jika tidak ada kondisi sebelumnya yang bernilai <code>True</code> . |
| <code>print("Pilihan tidak valid")</code> | Untuk mencetak dan memberi tahu pengguna bahwa pilihan yang mereka buat tidak sesuai dengan opsi yang diharapkan. Ini adalah |

| | |
|--|--|
| | pesan umum yang berguna untuk memberikan umpan balik kepada pengguna dalam kasus input yang tidak valid. |
| <code>print("Hasil:")</code> | Memberikan penanda bahwa hasil operasi atau output dari program akan dicetak setelah pesan tersebut. |
| <code>print_matrix(result)</code> | Mencetak hasil operasi matriks (penjumlahan atau pengurangan) ke layar. |
| <code>elif choice == '3'</code> | Menunjukkan bahwa program akan menjalankan serangkaian instruksi jika pengguna memilih menu dengan nomor 3. |
| <code>A = input_matrix(2)</code> | Untuk menginputkan nilai nilai elemen matriks. |
| <code>result = matrix_inverse(A)</code> | Digunakan untuk menghitung invers dari matriks A menggunakan fungsi <code>matrix_inverse()</code> dan mencetak hasilnya. |
| <code>if result is not None</code> | Untuk memeriksa apakah hasil dari perhitungan invers matriks (result) tidak bernilai None. |
| <code>print("Hasil:")</code> | Memberikan penanda bahwa hasil operasi atau output dari program akan dicetak setelah pesan tersebut. |
| <code>print_matrix(result)</code> | Digunakan untuk mencetak hasil matriks ke layar. |
| <code>else</code> | Untuk menangani situasi di mana kondisi pada pernyataan if atau elif sebelumnya tidak terpenuhi. Ini memberikan blok kode yang akan dieksekusi jika tidak ada kondisi sebelumnya yang bernilai True. |
| <code>print("Matriks tidak memiliki invers . ")</code> | Memberi tahu pengguna bahwa matriks yang dimasukkan tidak memiliki invers. |

| | |
|--|--|
| <code>elif choice == '4'</code> | Menunjukkan bahwa program akan menjalankan serangkaian instruksi jika pengguna memilih menu dengan nomor 4. |
| <code>determinant_menu()</code> | Untuk menampilkan menu pilihan untuk operasi determinan pada matriks. |
| <code>determinant_choice = input("Pilih operasi (1-2): ")</code> | Digunakan untuk meminta pengguna memilih jenis operasi determinan yang ingin dihitung, misalnya, determinan untuk matriks 2x2 atau 3x3. |
| <code>if determinant_choice == '1'</code> | Digunakan untuk menangani situasi di mana pengguna memilih operasi determinan untuk matriks 2x2. |
| <code>A = input_matrix(2)</code> | Memasukkan nilai-nilai elemen matriks. |
| <code>result = np.linalg.det(A)</code> | Digunakan untuk menghitung determinan dari matriks A menggunakan fungsi <code>np.linalg.det()</code> . Hasilnya kemudian dicetak ke layar menggunakan pernyataan <code>print</code> . |
| <code>elif determinant_choice == '2'</code> | Digunakan untuk menangani situasi di mana pengguna memilih operasi determinan untuk matriks 3x3. |
| <code>A = input_matrix(3)</code> | Memasukkan nilai-nilai elemen matriks. |
| <code>result = np.linalg.det(A)</code> | Digunakan untuk menghitung determinan dari matriks A menggunakan fungsi <code>np.linalg.det()</code> . Hasilnya kemudian dicetak ke layar menggunakan pernyataan <code>print</code> . |
| <code>else</code> | Untuk menangani situasi di mana kondisi pada pernyataan <code>if</code> atau <code>elif</code> sebelumnya tidak terpenuhi. Ini memberikan blok kode yang akan dieksekusi jika tidak ada kondisi sebelumnya yang bernilai <code>True</code> . |

| | |
|---|---|
| <code>print("Pilihan tidak valid")</code> | Untuk mencetak dan memberi tahu pengguna bahwa pilihan yang mereka buat tidak sesuai dengan opsi yang diharapkan. Ini adalah pesan umum yang berguna untuk memberikan umpan balik kepada pengguna dalam kasus input yang tidak valid. |
| <code>print(f"Hasil determinan: {result}")</code> | Digunakan untuk mencetak hasil perhitungan determinan matriks ke layar. |
| <code>elif choice == '5'</code> | Menunjukkan bahwa program akan menjalankan serangkaian instruksi jika pengguna memilih menu dengan nomor 5. |
| <code>A, b = linear_system_solution_menu()</code> | Digunakan untuk memanggil fungsi <code>linear_system_solution_menu()</code> dan mengambil nilai yang dikembalikan oleh fungsi tersebut. fungsi tersebut meminta pengguna untuk memasukkan koefisien matriks A dan vektor B dalam sistem persamaan linier. |
| <code>try</code> | Digunakan untuk memulai blok yang akan diuji apakah terjadi pengecualian (exception). |
| <code>result = np.linalg.solve(A, b)</code> | Untuk mencari solusi dari sistem persamaan linier yang diberikan oleh matriks A dan vektor b. Hasilnya kemudian dicetak ke layar menggunakan pernyataan <code>print</code> . |
| <code>print("Hasil:")</code> | Memberikan penanda bahwa hasil operasi atau output dari program akan dicetak setelah pesan tersebut. |
| <code>print_matrix(result)</code> | Digunakan untuk mencetak hasil matriks ke layar. |
| <code>except np.linalg.LinAlgError</code> | Untuk menangani pengecualian khusus yang mungkin terjadi selama eksekusi blok <code>try</code> . |

| | |
|---|--|
| <code>print("Sistem persamaan linier tidak memiliki solusi unik.")</code> | Program akan mencetak pesan yang memberitahu pengguna bahwa sistem persamaan linier tidak memiliki solusi unik. |
| <code>elif choice == '6'</code> | Menunjukkan bahwa program akan menjalankan serangkaian instruksi jika pengguna memilih menu dengan nomor 6. |
| <code>print("Program selesai.")</code> | Program akan mencetak pesan yang memberitahu pengguna bahwa program selesai. |
| <code>break</code> | Digunakan untuk keluar dari loop utama, mengakhiri eksekusi program. |
| <code>else</code> | Untuk menangani situasi di mana kondisi pada pernyataan if atau elif sebelumnya tidak terpenuhi. Ini memberikan blok kode yang akan dieksekusi jika tidak ada kondisi sebelumnya yang bernilai True. |
| <code>print("Pilihan tidak valid. Silakan pilih angka 1-6.")</code> | Untuk mencetak dan memberi tahu pengguna bahwa pilihan yang mereka buat tidak sesuai dengan opsi yang diharapkan, lalu pilih input 1-6. |

BAB IV

PENGUJIAN

Tampilan menu pilih program yang ingin di jalankan .

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

Misalkan memilih nomor 1 maka akan muncul tampilan seperti dibawah ini

Penjumlahan Matriks dan Pengurangan Matriks

- Penjumlahan matriks

Studi kasus

$$\begin{bmatrix} 2 & 4 \\ 8 & 4 \end{bmatrix} + \begin{bmatrix} 2 & 4 \\ 8 & 4 \end{bmatrix} =$$

```

1
PILIHAN
1. Penjumlahan matriks
2. Pengurangan matriks
Pilih operasi (1-2):
1
```

Selanjutnya pilih submenu yang ingin dijalankan apabila memilih satu maka program akan menjalankan penjumlahan matriks. Lalu masukan elemen pada penjumlahan matriks dibawah

```
Masukkan nilai matriks 2:
Masukkan nilai a11:
2
Masukkan nilai a12:
4
Masukkan nilai a21:
8
Masukkan nilai a22:
4
Masukkan nilai matriks 2:
Masukkan nilai a11:
2
Masukkan nilai a12:
4
Masukkan nilai a21:
8
Masukkan nilai a22:
4
```

Setelah memasukan elemen maka akan mncul hasil seperti dibawah ini

```
Hasil:
[4. 8.]
[16. 8.]
```

- Pengurangan matriks

Studi kasus

$$\begin{bmatrix} 2 & 4 \\ 8 & 4 \end{bmatrix} - \begin{bmatrix} 2 & 4 \\ 8 & 4 \end{bmatrix} =$$

Selanjutnya untuk memunculkan pengurangan matriks pilih 1

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

Lalu akan muncul tampilan seperti dibawah lalu pilih submenu kedua yaitu pengurangan matriks

```
Pilih menu (1-6):
1
PILIHAN
1. Penjumlahan matriks
2. Pengurangan matriks
Pilih operasi (1-2):
2
```

Selanjutnya pengguna akan diminta untuk memasukan elemem pada pengurangan matriks dibawah

```
Masukkan nilai matriks 2:
Masukkan nilai a11:
2
Masukkan nilai a12:
4
Masukkan nilai a21:
8
Masukkan nilai a22:
4
Masukkan nilai matriks 2:
Masukkan nilai a11:
2
Masukkan nilai a12:
4
Masukkan nilai a21:
8
Masukkan nilai a22:
4
```

Maka hasil yang di dapatkan dari pengurangan matriks diatas adalah sebagai berikut

```
Hasil:
[0. 0.]
[0. 0.]
```

Matriks Transpose

- Matriks Transpose Ordo 2x2

Studi kasus :

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

Selanjutnya kita akan menghitung matriks transpose, maka pada menu dibawah pengguna bisa memilih menu kedua yaitu matriks transpose

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

Maka akan muncul tampilan dibawah, untuk menampilkan matriks dengan ordo 2x2 maka pengguna bisa memilih opsi 1

```
Pilih menu (1-6):
2
PILIHAN
1. Matriks 2x2
2. Matriks 3x3
Pilih operasi (1-2):
1
```

Selanjutnya akan muncul tampilan dibawah, pengguna diperintahkan memasukan elemen kedalam rumus matriks ordo 2x2

```
Masukkan nilai matriks 2:
Masukkan nilai a11:
1
Masukkan nilai a12:
2
Masukkan nilai a21:
3
Masukkan nilai a22:
4
```

Hasil matriks transpose berordo 2x2 adalah sebagai berikut

```
Hasil:
[1. 3.]
[2. 4.]
```

- Matriks Transpose Ordo 3x3

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Selanjutnya kita akan menghitung matriks transpose berordo 3x3, maka pada menu dibawah pengguna bisa memilih menu kedua yaitu matriks transpose

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

Lalu akan muncul pilihan submenu seperti dibawah dan pilih matriks 3x3

```
Pilih menu (1-6):  
2  
PILIHAN  
1. Matriks 2x2  
2. Matriks 3x3  
Pilih operasi (1-2):  
2
```

Kemudian program akan menampilkan perintah untuk memasukan elemen matriks

```
Masukkan nilai matriks 3:  
Masukkan nilai a11:  
1  
Masukkan nilai a12:  
2  
Masukkan nilai a13:  
3  
Masukkan nilai a21:  
4  
Masukkan nilai a22:  
5  
Masukkan nilai a23:  
6  
Masukkan nilai a31:  
7  
Masukkan nilai a32:  
8  
Masukkan nilai a33:  
9
```

Setelah memasukan elemen maka akan muncul hasil seperti pada gambar dibawah

```
Hasil:  
[1. 4. 7.]  
[2. 5. 8.]  
[3. 6. 9.]
```

Matriks Balikan

Selanjutnya kita akan menghitung matriks balikan, maka pada menu dibawah pengguna bisa memilih menu ketiga yaitu matriks balikan

Studi kasus :

$$\begin{bmatrix} 9 & 8 \\ 6 & 5 \end{bmatrix}$$

```
MENU
1. Penjumlahan dan Pengurangan
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
3
```

Selanjutnya program akan meminta untuk memasukan elemen kedalam matriks balikan

```
Masukkan nilai matriks 2:
Masukkan nilai a11:
9
Masukkan nilai a12:
8
Masukkan nilai a21:
6
Masukkan nilai a22:
5
```

Program akan menampilkan hasil matriks balikan seperti berikut

```
Hasil:
[-1.66666667  2.66666667]
[ 2. -3.]
```

Determinan matriks

- Determinan matriks 2x2

Selanjutnya kita akan menghitung Determinan, maka pada menu dibawah pengguna bisa memilih menu keempat yaitu Determinan

Studi kasus :

$$\begin{bmatrix} 2 & 1 \\ 2 & 6 \end{bmatrix}$$

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

Lalu akan muncul pilihan submenu seperti dibawah dan pilih matriks 2x2

```
Pilih menu (1-6):
4
PILIHAN
1. Matriks 2x2
2. Matriks 3x3
Pilih operasi (1-2):
1
```

Selanjutnya program akan meminta untuk memasukan elemen kedalam determinan matriks 2x2

```
Masukkan nilai matriks 2:
Masukkan nilai a11:
2
Masukkan nilai a12:
1
Masukkan nilai a21:
2
Masukkan nilai a22:
0
```

Program akan menampilkan hasil matriks balikan seperti berikut

```
Hasil determinan: -2.0
```

- Determinan matriks 3x3

Selanjutnya kita akan menghitung Determinan, maka pada menu dibawah pengguna bisa memilih menu keempat yaitu Determinan

Studi kasus:

$$\begin{bmatrix} 2 & 1 & 3 \\ 0 & 1 & 2 \\ 3 & 2 & 0 \end{bmatrix}$$

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

Lalu akan muncul pilihan submenu seperti dibawah dan pilih matriks 3x3

```
Pilih menu (1-6):
4
PILIHAN
1. Matriks 2x2
2. Matriks 3x3
Pilih operasi (1-2):
2
```

Selanjutnya program akan meminta untuk memasukan elemen kedalam determinan matriks 3x3

```
Masukkan nilai matriks 3:
Masukkan nilai a11:
2
Masukkan nilai a12:
1
Masukkan nilai a13:
3
Masukkan nilai a21:
0
Masukkan nilai a22:
1
Masukkan nilai a23:
2
Masukkan nilai a31:
3
Masukkan nilai a32:
2
Masukkan nilai a33:
0
```

Program akan menampilkan hasil matriks balikan seperti berikut

```
Hasil determinan: -11.000000000000002
```

Sistem Persaman Linier

Selanjutnya kita akan menghitung Sistem Persamaan Linier, maka pada menu dibawah pengguna bisa memilih menu kelima

Studi kasus:

$$\begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix}$$

```
MENU
1. Penjumlahan dan Pengurangan Matriks
2. Matriks Transpose
3. Matriks Balikan
4. Determinan
5. Sistem Persamaan Linier
6. Keluar
Pilih menu (1-6):
```

```
Pilih menu (1-6):  
5  
Masukkan koefisien matriks A dan vektor b:  
Masukkan nilai matriks 2:
```

Selanjutnya program akan meminta untuk memasukan elemen kedalam Sistem Persamaan Linier

```
Masukkan nilai a11:  
1  
Masukkan nilai a12:  
2  
Masukkan nilai a21:  
0  
Masukkan nilai a22:  
0  
Masukkan nilai matriks 2:  
Masukkan nilai a11:  
1  
Masukkan nilai a12:  
2  
Masukkan nilai a21:  
0  
Masukkan nilai a22:  
0
```

Maka hasilnya :

```
Sistem persamaan linier tidak memiliki solusi unik.
```

keluar

```
MENU  
1. Penjumlahan dan Pengurangan Matriks  
2. Matriks Transpose  
3. Matriks Balikan  
4. Determinan  
5. Sistem Persamaan Linier  
6. Keluar  
Pilih menu (1-6):
```



```
Pilih menu (1-6):  
6  
Program selesai.
```

BAB V

KESIMPULAN

A. Kesimpulan

1. Kami telah berhasil membuat sebuah program yang mampu untuk menyelesaikan suatu Sistem persamaan linier yang dapat memilah SPL dengan solusi unik, solusi banyak, dan tidak ada solusi.
2. Kami telah berhasil membuat sebuah program yang mampu menghitung matriks transpose dan determinan dengan Ordo 2×2 dan Ordo 3×3
3. Kami telah berhasil membuat sebuah program matriks balikan
4. Kami telah berhasil membuat sebuah program yang mampu menyelesaikan persoalan penjumlahan dan pengurangan matriks.

B. Saran

Dikarenakan pada tugas kali ini hanya diberi waktu 3 minggu, maka Agak lama bagi kami untuk belajar bahasa pemrograman python dan langsung mempraktikannya ke tugas besar (program yang lumayan besar) apalagi jika harus dibuat dengan GUI. Kedepannya (jika memungkinkan), jika waktunya ditambah lebih lama lagi, maka mungkin dapat untuk mengembangkan kemampuan kami menjadi lebih baik.

C. Refleksi

Pada tugas besar kali ini, kami menemui kendala, yakni beberapa dari kami baru saja mengenal GUI, dan harus mengerjakan tugas sambil belajar sekaligus, jadi lumayan banyak debug yang harus dilakukan.

DAFTAR PUSTAKA

<https://www.detik.com/edu/detikpedia/d-5936159/determinan-matriks-penjelasan-dan-contoh-soal>

<https://gdscuniversitasmulia.medium.com/eliminasi-gauss-jordan-79b87e67befd>

<https://katadata.co.id/intan/lifestyle/63f8b44a67a6f/memahami-metode-eliminasi-gauss-dan-pembahasan-soal>

<https://katadata.co.id/agung/berita/62e8d6565b11d/memahami-persamaan-linear-pengertian-jenis-dan-contoh-soalnya>