

LAPORAN TUGAS BESAR 01

IF 2123 ALJABAR LINEAR DAN GEOMETRI

Kelompok Jun Hok 88



Disusun oleh:

Elbert Chailes	13522045
Kharris Khisunica	13522051
Benardo	13522055

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BAB 1

DESKRIPSI MASALAH

Sistem persamaan linier (SPL) banyak ditemukan di dalam bidang sains dan rekayasa. Anda sudah mempelajari berbagai metode untuk menyelesaikan SPL, termasuk menghitung determinan matriks. Sembarang SPL dapat diselesaikan dengan beberapa metode, yaitu metode eliminasi Gauss, metode eliminasi Gauss-Jordan, metode matriks balikan ($x = A^{-1}b$), dan kaidah *Cramer* (khusus untuk SPL dengan n peubah dan n persamaan). Solusi sebuah SPL mungkin tidak ada, banyak (tidak berhingga), atau hanya satu (unik/tunggal). Di dalam Tugas Besar 1 ini, Anda diminta membuat satu atau lebih *library* aljabar linier dalam Bahasa Java. Library tersebut berisi fungsi-fungsi seperti eliminasi Gauss, eliminasi Gauss-Jordan, menentukan balikan matriks, menghitung determinan, kaidah Cramer (kaidah Cramer khusus untuk SPL dengan n peubah dan n persamaan). Selanjutnya, gunakan *library* tersebut di dalam program Java untuk menyelesaikan berbagai persoalan yang dimodelkan dalam bentuk SPL, menyelesaikan persoalan interpolasi, dan persoalan regresi.

1. Tujuan

1. Menentukan solusi dari sistem persamaan linear (SPL) yang diberikan dengan metode eliminasi Gauss, metode eliminasi Gauss-Jordan, metode matriks balikan, dan kaidah Cramer.
2. Menghitung determinan matriks dengan metode reduksi baris dan metode ekspansi kofaktor
3. Menentukan balikan matriks dengan metode adjoint matrix dan metode matriks identitas.
4. Menyelesaikan persoalan interpolasi polinomial, interpolasi regresi linear, dan interpolasi *bicubic spline*.

2. Spesifikasi

1. Program dapat menerima masukan (*input*) baik dari *keyboard* maupun membaca masukan dari *file text*. Untuk SPL, masukan dari *keyboard* adalah m , n , koefisien a_{ij} , dan b_i . Masukan dari *file* berbentuk matriks *augmented* tanpa tanda kurung, setiap elemen matriks dipisah oleh spasi. Misalnya,

```
3 4.5 2.8 10 12
-3 7 8.3 11 -4
0.5 -10 -9 12 0
```

2. Untuk persoalan menghitung determinan dan matriks balikan, masukan dari *keyboard* adalah n dan koefisien a_{ij} . Masukan dari *file* berbentuk matriks, setiap elemen matriks dipisah oleh spasi. Misalnya,

```
3 4.5 2.8
-3 7 8.3
0.5 -10 -9
```

Luaran (*output*) disesuaikan dengan persoalan (determinan atau invers) dan penghitungan balikan/invers dilakukan dengan metode matriks balikan dan adjoin.

3. Untuk persoalan interpolasi, masukannya jika dari *keyboard* adalah n , (x_0, y_0) , (x_1, y_1) , ..., (x_n, y_n) , dan nilai x yang akan ditaksir nilai fungsinya. Jika masukannya dari *file*, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung. Masukan kemudian dilanjutkan dengan satu buah baris berisi satu buah nilai x yang akan ditaksir menggunakan fungsi interpolasi yang telah didefinisikan. Misalnya jika titik-titik datanya adalah $(8.0, 2.0794)$, $(9.0, 2.1972)$, dan $(9.5, 2.2513)$ dan akan mencari nilai y saat $x = 8.3$, maka di dalam *file text* ditulis sebagai berikut:

```
8.0 2.0794
9.0 2.1972
9.5 2.2513
8.3
```

4. Untuk persoalan regresi, masukannya jika dari *keyboard* adalah n (jumlah peubah x), m (jumlah sampel), semua nilai-nilai x_{1i} , x_{2i} , ..., x_{ni} , nilai y_i , dan nilai-nilai x_k yang akan ditaksir nilai fungsinya. Jika masukannya dari *file*, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung.
5. Untuk persoalan SPL, luaran 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 $x_1 = t$).
6. Untuk persoalan polinom interpolasi dan regresi, luarannya adalah persamaan polinom/regresi dan taksiran nilai fungsi pada x yang diberikan. Contoh luaran untuk interpolasi adalah

$$f(x) = -0.0064x^2 + 0.2266x + 0.6762, \quad f(5) = \dots$$

dan untuk regresi adalah

$$f(x) = -9.5872 + 1.0732x_1, \quad f(x_k) = \dots$$

7. Untuk persoalan *bicubic spline interpolation*, masukan dari *file text* (.txt) yang berisi matriks berukuran 4×4 yang berisi konfigurasi nilai fungsi dan turunan berarah disekitarnya, diikuti dengan nilai a dan b untuk mencari nilai $f(a, b)$.

Misalnya jika nilai dari $f(0, 0)$, $f(1, 0)$, $f(0, 1)$, $f(1, 1)$, $f_x(0, 0)$, $f_x(1, 0)$, $f_x(0, 1)$, $f_x(1, 1)$, $f_y(0, 0)$, $f_y(1, 0)$, $f_y(0, 1)$, $f_y(1, 1)$, $f_{xy}(0, 0)$, $f_{xy}(1, 0)$, $f_{xy}(0, 1)$, $f_{xy}(1, 1)$ berturut-turut adalah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 serta nilai a dan b yang dicari berturut-turut adalah 0.5 dan 0.5 maka isi *file text* ditulis sebagai berikut:

```
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
0.5 0.5
```

Luaran yang dihasilkan adalah nilai dari $f(0.5, 0.5)$.

8. Luaran program harus dapat ditampilkan **pada layar komputer dan dapat disimpan ke dalam file**.

9. Bahasa program yang digunakan adalah Java. Anda bebas untuk menggunakan versi java apapun dengan catatan di atas java versi 8 (8/9/11/15/17/19/20).
10. Program **tidak harus** berbasis GUI, cukup *text-based* saja, namun boleh menggunakan GUI (memakai kakas *Eclipse* misalnya).
11. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

MENU

1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic Spline
6. Regresi linier berganda
7. Keluar

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

1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer

Begitu juga untuk pilihan menu nomor 2 dan 3.

BAB 2

TEORI SINGKAT

1. Metode Operasi Baris Elementer (OBE)

Operasi Baris Elementer adalah suatu operasi yang digunakan terhadap matriks dengan tujuan yang beragam, meliputi untuk memperoleh penyelesaian dari permasalahan Sistem Persamaan Linear (SPL), untuk menentukan balikan dari sebuah matriks, atau bahkan menentukan nilai determinan sebuah matriks. Untuk mendapatkan solusi dari SPL, OBE dilakukan pada *augmented matrix* hingga terbentuk matriks eselon baris (matriks yang memiliki 1 utama pada setiap baris, kecuali baris yang seluruhnya nol) yang berujung penyelesaian SPL dengan metode eliminasi Gauss atau matriks eselon baris tereduksi (matriks eselon baris dengan setiap kolom yang memiliki 1 utama memiliki nol di tempat lain) yang berujung pada penyelesaian SPL dengan metode eliminasi Gauss-Jordan. Adapun tiga OBE terhadap *augmented matrix* :

1. Kalikan sebuah baris dengan konstanta tidak nol.
2. Pertukarkan dua buah baris.
3. Tambahkan sebuah baris dengan kelipatan baris lainnya.

2. Metode Eliminasi Gauss

Dalam matematika, eliminasi Gauss adalah algoritma yang digunakan untuk menyelesaikan sistem persamaan linear (SPL). Mekanisme eliminasi Gauss ini dipopulerkan oleh seorang matematikawan yang bernama Carl Friedrich Gauss (1777-1855). Mekanisme eliminasi Gauss dalam mencari penyelesaian solusi atas sebuah SPL adalah sebagai berikut.

1. Nyatakan SPL dalam bentuk *augmented matrix*
2. Terapkan OBE pada *augmented matrix* hingga terbentuk matriks eselon baris
3. Selesaikan persamaan yang berkoresponden pada matriks eselon baris dengan teknik substitusi balik (*backward substitution*).

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

3. Metode Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan merupakan pengembangan metode eliminasi Gauss. OBE diterapkan pada *augmented matrix* sehingga menghasilkan matriks eselon baris tereduksi. Maka, tidak perlu dilakukan lagi substitusi mundur (*backward substitution*) untuk memperoleh nilai-nilai variabel, melainkan langsung diperoleh dari *augmented matrix* jika solusinya unik.

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

Dalam pelaksanaan mekanisme metode eliminasi Gauss-Jordan, terdapat dua fase yang terjadi, yaitu:

1. Fase maju (*forward phase*) atau fase eliminasi Gauss untuk menghasilkan nilai-nilai 0 dibawah 1 utama.

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

2. Fase mundur (*backward phase*) untuk menghasilkan nilai-nilai 0 di atas satu utama.

$$\begin{bmatrix} 1 & 3/2 & -1/2 & 5/2 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix} \xrightarrow{\text{R1} - (3/2)\text{R2}} \begin{bmatrix} 1 & 0 & -5/4 & -11/4 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix} \xrightarrow{\begin{matrix} \text{R1} + (5/4)\text{R3} \\ \text{R2} - (1/2)\text{R3} \end{matrix}} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

Matrks eselon baris tereduksi

Kedua fase dapat dilakukan secara bersamaan ataupun sekuensial.

4. Determinan

Determinan adalah nilai yang dihitung dari operasi unsur-unsur suatu matriks persegi (matriks yang memiliki jumlah baris dan kolom sama). Determinan suatu matriks

dapat dihitung dengan berbagai metode, seperti metode reduksi baris dan metode ekspansi kofaktor.

Dengan metode reduksi baris, determinan sebuah matriks dapat diperoleh dengan melakukan OBE hingga diperoleh matriks segitiga, baik segitiga atas maupun segitiga bawah seperti gambar berikut.

$$[A] \stackrel{\text{OBE}}{\sim} [\text{matriks segitiga bawah}]$$

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \stackrel{\text{OBE}}{\sim} \begin{bmatrix} a'_{11} & a'_{12} & \dots & a'_{1n} \\ 0 & a'_{22} & \dots & a'_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & a'_{nn} \end{bmatrix}$$

$$\text{maka } \det(A) = (-1)^p a'_{11} a'_{22} \dots a'_{nn} \quad *)$$

p menyatakan banyaknya operasi pertukaran baris di dalam OBE

*) Asumsi tidak ada operasi perkalian baris dengan konstanta k

Jika selama reduksi baris terdapat OBE berupa perkalian baris-baris matriks dengan k_1, k_2, \dots, k_m , maka berlaku seperti berikut.

$$\det(A) = \frac{(-1)^p a'_{11} a'_{22} \dots a'_{nn}}{k_1 k_2 \dots k_m}$$

Dengan metode ekspansi kofaktor, misalkan matriks A berukuran $n \times n$ seperti berikut.

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

Didefinisikan :

M_{ij} = minor entri a_{ij} = determinan upa-matriks (*submatrix*) yang elemen-elemennya tidak berada pada baris i dan kolom j

$C_{ij} = (-1)^{i+j} M_{ij}$ = kofaktor entri a_{ij}

Dengan menggunakan kofaktor, maka determinan matriks A dapat dihitung dengan salah satu dari persamaan berikut:

$$\begin{aligned}\det(A) &= a_{11}C_{11} + a_{12}C_{12} + \dots + a_{1n}C_{1n} \\ \det(A) &= a_{21}C_{21} + a_{22}C_{22} + \dots + a_{2n}C_{2n} \\ &\vdots \\ \det(A) &= a_{n1}C_{n1} + a_{n2}C_{n2} + \dots + a_{nn}C_{nn}\end{aligned}$$

Secara baris

$$\begin{aligned}\det(A) &= a_{11}C_{11} + a_{21}C_{21} + \dots + a_{n1}C_{n1} \\ \det(A) &= a_{12}C_{12} + a_{22}C_{22} + \dots + a_{n2}C_{n2} \\ &\vdots \\ \det(A) &= a_{1n}C_{1n} + a_{2n}C_{2n} + \dots + a_{nn}C_{nn}\end{aligned}$$

Secara kolom

5. Matriks Balikan

Matriks balikan adalah kebalikan dari sebuah matriks yang apabila matriks tersebut dikalikan dengan balikkannya, akan menjadi matriks identitas. Matriks yang memiliki matriks balikan sudah pasti merupakan matriks persegi. Matriks balikan dari matriks A merupakan $(A)^{-1}$ sehingga dapat ditulis $A(A)^{-1} = (A)^{-1}A = I$. Matriks balikan juga dapat dicari dengan metode eliminasi Gauss-Jordan dan metode adjoin matriks.

Dengan metode eliminasi Gauss-Jordan, matriks balikan dapat dicari dengan cara berikut:

$$[A|I] \xrightarrow{\text{G-J}} [I|A^{-1}]$$

Jika ditemukan baris yang seluruh elemennya adalah 0 ketika dilakukan operasi dengan eliminasi Gauss-Jordan, maka matriks tersebut dapat dipastikan tidak memiliki balikan.

Dengan metode adjoin matriks juga dapat dicari balikan dari sebuah matriks dengan rumus sebagai berikut:

$$A^{-1} = \frac{1}{\det(A)} \text{adj}(A)$$

Jika determinan dari sebuah matriks adalah nol, maka sudah dapat dipastikan bahwa matriks tersebut tidak memiliki balikan.

Dengan adanya matriks balikan ini, dapat diperoleh solusi dari SPL $Ax = b$ dengan mengubah persamaan tersebut menjadi $x = (A)^{-1} B$ sehingga akan diperoleh solusi berupa matriks solusi dari x .

6. Matriks Kofaktor

Matriks kofaktor merupakan matriks yang terdiri dari kofaktor-kofaktor matriks itu sendiri. Misalkan didefinisikan matriks A adalah matriks $n \times n$ dan C_{ij} adalah kofaktor entri a_{ij} , maka matriks kofaktor dari A adalah sebagai berikut.

$$\begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix}$$

7. Adjoin Matriks

Adjoin matriks merupakan transpose dari suatu matriks yang elemen-elemennya merupakan kofaktor dari elemen-elemen matriks tersebut, dengan kata lain transpose dari matriks kofaktornya. Maka dari itu, misalkan didefinisikan matriks A adalah matriks $n \times n$, adjoin matriks A dapat ditulis sebagai berikut:

$$\text{adj}(A) = \text{transpose matriks kofaktor dari } A$$

8. Kaidah Cramer

Kaidah *Cramer* merupakan rumus yang dapat digunakan untuk menyelesaikan sistem persamaan linear dengan banyak persamaan sama dengan banyak variabel, dan berlaku ketika sistem tersebut memiliki solusi yang tunggal. Kaidah ini menyatakan solusi dengan menggunakan determinan matriks koefisien dan determinan matriks lain yang diperoleh dengan mengganti salah satu kolom matriks koefisien dengan vektor yang berada pada sebelah kanan persamaan. Kaidah ini dipopulerkan oleh seorang matematikawan bernama Swiss Gabriel Cramer (1704-1752).

Jika $Ax = b$ adalah SPL yang terdiri dari n persamaan linier dengan n peubah (variabel) sedemikian sehingga determinan A tidak sama dengan nol, maka SPL tersebut memiliki solusi yang unik, yaitu:

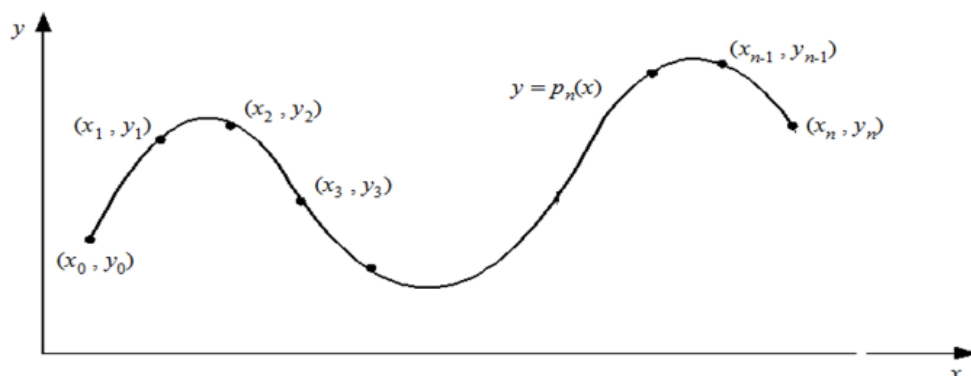
$$x_1 = \frac{\det(A_1)}{\det(A)}, \quad x_2 = \frac{\det(A_2)}{\det(A)}, \quad \dots, \quad x_n = \frac{\det(A_n)}{\det(A)}$$

yang dalam hal ini, A_j adalah matriks yang diperoleh dengan mengganti entri pada kolom ke- j dari A dengan entri dari matriks

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

9. Interpolasi Polinomial

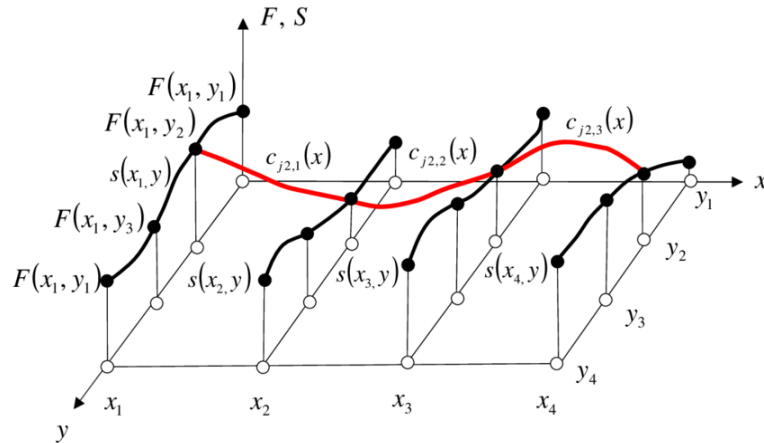
Interpolasi polinomial merupakan teknik interpolasi dengan mengasumsikan pola data yang dipakai mengikuti pola polinomial baik berderajat satu (linier) maupun berderajat tinggi untuk kemudian membentuk persamaan polinomial dari data tersebut. Persamaan polinomial yang terbentuk kemudian dapat digunakan untuk melakukan interpolasi dari nilai yang diketahui maupun melakukan ekstrapolasi (tafsiran) nilai yang berada di luar rentang data yang diketahui.



10. Interpolasi bicubic spline

Interpolasi *bicubic spline* adalah metode interpolasi yang digunakan untuk mengaproksimasi fungsi di antara titik-titik data yang diketahui. Bicubic spline

interpolation melibatkan konsep spline dan konstruksi serangkaian polinomial kubik di dalam setiap sel segi empat dari data yang diberikan. Pendekatan ini menciptakan permukaan yang halus dan kontinu, memungkinkan untuk perluasan data secara visual yang lebih akurat daripada metode interpolasi linear.



Implementasi interpolasi ini biasanya digunakan sebagai teknik interpolasi pada data 2D, seperti dalam pembesaran citra.

11. Regresi Linear Berganda

Regresi linear berganda merupakan model regresi yang melibatkan lebih dari satu variabel independen. Regresi linear berganda ini merupakan salah satu metode untuk memprediksi nilai berdasarkan data yang dimiliki selain metode interpolasi polinomial. Persamaan umum dari regresi linear berganda, yaitu:

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_k x_{ki} + \epsilon_i$$

Nilai dari setiap β_i dapat ditentukan dengan menggunakan *Normal Estimation Equation for Multiple Linear Regression* berikut.

$$\begin{aligned} nb_0 + b_1 \sum_{i=1}^n x_{1i} + b_2 \sum_{i=1}^n x_{2i} + \cdots + b_k \sum_{i=1}^n x_{ki} &= \sum_{i=1}^n y_i \\ b_0 \sum_{i=1}^n x_{1i} + b_1 \sum_{i=1}^n x_{1i}^2 + b_2 \sum_{i=1}^n x_{1i}x_{2i} + \cdots + b_k \sum_{i=1}^n x_{1i}x_{ki} &= \sum_{i=1}^n x_{1i}y_i \\ \vdots & \\ b_0 \sum_{i=1}^n x_{ki} + b_1 \sum_{i=1}^n x_{ki}x_{1i} + b_2 \sum_{i=1}^n x_{ki}x_{2i} + \cdots + b_k \sum_{i=1}^n x_{ki}^2 &= \sum_{i=1}^n x_{ki}y_i \end{aligned}$$

BAB 3

IMPLEMENTASI PROGRAM

1. Matrix.java

Class ini berisi seluruh fungsi dan prosedur yang digunakan untuk menyelesaikan semua spesifikasi masalah yang diberikan, baik pencarian solusi sistem persamaan linear dengan metode eliminasi Gauss, metode eliminasi Gauss-Jordan, metode *cramer*, metode matriks balikan, pencarian nilai determinan daripada matriks dengan menggunakan metode reduksi baris dan metode ekspansi kofaktor, pencarian balikan dari matriks dengan menggunakan metode adjoin matriks dan metode matriks identitas, menyelesaikan persoalan interpolasi polinomial, regresi linear berganda, dan interpolasi *bicubic spline*.

- **Atribut**

Atribut	Deskripsi
public double[][] data	Menampung data dengan tipe double yang merupakan isi dari matriks
public int jumlahBaris	Berisi data dengan tipe integer yang menampung jumlah baris matriks
public int jumlahKolom	Berisi data dengan tipe integer yang menampung jumlah kolom matriks

- **Konstruktor**

Konstruktor	Deskripsi
public Matrix()	Konstruktor untuk membuat matriks dengan isi bertipe double yang memiliki jumlah baris sebanyak 0 dan jumlah kolom sebanyak 0
public Matrix(int rows, int cols)	Konstruktor untuk membuat matriks dengan isi bertipe double serta memiliki jumlah baris sebanyak rows dan jumlah kolom sebanyak cols

- **Fungsi / Prosedur**

Fungsi / Prosedur	Deskripsi
public void printMatriks()	Menampilkan isi matriks ke layar
public void bacaMatriks(Scanner scanner)	Meminta dan menyimpan isian matriks yang dimasukkan oleh user ke dalam matriks
public void bacaMatriksDariFile(String filename)	Membaca dan menyimpan isian matriks yang terdapat pada file <i>filename</i>
public double getElmt(int i, int j)	Mengembalikan isi data matriks pada baris i dan kolom j
public void setElmt(int i, int j, double val)	Melakukan modifikasi isi data matriks pada baris i dan kolom j
public int getBaris()	Mengembalikan nilai jumlah baris yang dimiliki oleh matriks
public int getKolom()	Mengembalikan nilai jumlah kolom yang dimiliki oleh matriks
public boolean isNoSolution(int i)	Mengembalikan true jika hasil obe tidak mempunyai solusi, dan mengembalikan false jika ada solusi
public boolean isAllZero(int i)	Mengembalikan true jika elemen dalam baris i semuanya bernilai nol, dan mengembalikan false jika tidak semua elemen di baris i semuanya bernilai nol
public Matrix makeItSquare()	Mengembalikan matriks NxN atau matriks koefisien variabel dari matriks <i>augmented</i> pada SPL, terutama metode <i>cramer</i> .
public Matrix copyMatrix()	Mengembalikan matriks sama persis dengan matriks yang di- <i>copy</i>
public static Matrix multiplyMatrix(Matrix m1, Matrix m2)	Mengembalikan matriks hasil perkalian matriks antara matriks m1 dan matriks m2
public Matrix changeRow(int i1, int i2)	Mengembalikan matriks setelah dilakukan pertukaran baris antara baris dengan indeks i1 dan indeks i2
public pchangeRow(int row1, int	Melakukan modifikasi langsung

row2)	terhadap matriks berupa pertukaran baris antara baris dengan indeks i1 dan indeks i2
public Matrix changeCol (int col1, int col2)	Mengembalikan matriks setelah dilakukan pertukaran kolom antara kolom dengan indeks col1 dan indeks col2
public boolean isSquareMatrix()	Mengembalikan true jika matriks memiliki baris dan kolom yang sama atau matriks merupakan <i>square matrix</i> dan false jika sebaliknya
public Matrix subMatrix(int a, int b)	Mengembalikan submatriks dari matriks dengan menghilangkan baris dengan indeks a dan kolom dengan indeks b sehingga submatriks memiliki ukuran baris matriks -1 dan ukuran kolom matriks - 1
public Matrix adjoinMatrix()	Mengembalikan hasil adjoin dari sebuah matriks
public double determinantWithCofExpansion()	Mengembalikan nilai dengan tipe double yang berupa nilai determinan dari sebuah matriks dengan metode ekspansi kofaktor. Jika matriks bukan merupakan matriks persegi maka fungsi akan mengembalikan nilai NaN, sedangkan jika sebaliknya akan mengembalikan nilai determinan matriks yang sebenarnya.
public double determinantWithReduksiBaris()	Mengembalikan nilai dengan tipe double yang berupa nilai determinan dari sebuah matriks dengan metode reduksi baris. Jika matriks bukan merupakan matriks persegi maka fungsi akan mengembalikan nilai NaN. Jika matriks merupakan matriks persegi, maka fungsi akan berusaha untuk mengubah program menjadi matriks segitiga bawah sehingga nilai determinan hanya berupa hasil kali dari diagonal matriks hasil reduksi baris.
public void	Prosedur ini mengeluarkan hasil

SPLwithCramerMethod(Scanner scanner)	penyelesaian dari SPL yang telah dikonversi menjadi bentuk <i>augmented matrix</i> . Jika matriks koefisien variabel bukan merupakan matriks persegi maka prosedur akan mengeluarkan output bahwa SPL tidak bisa diselesaikan dengan metode cramer. Jika matriks koefisien variabel persegi, dan determinannya bernilai 0 maka SPL akan tidak memiliki solusi. Jika koefisien variabel persegi, dan determinannya tidak sama dengan nol, maka solusi akan ditampilkan di layar user.
public Matrix gauss()	Mengembalikan Matriks eselon dari matriks yang dilakukan operasi
public Matrix gaussJordan()	Mengembalikan Matriks eselon tereduksi dari matriks yang dilakukan operasi
public void SPLwithInverseMethod(Scanner scanner)	Prosedur ini mengeluarkan hasil penyelesaian dari SPL yang telah dikonversi menjadi bentuk <i>augmented matrix</i> . Jika matriks koefisien variabel bukan merupakan matriks persegi maka prosedur akan mengeluarkan output bahwa SPL tidak dapat diselesaikan dengan metode balikan. Jika matriks koefisien variabel persegi, dan determinannya bernilai 0 maka SPL akan tidak memiliki solusi. Jika koefisien variabel persegi, dan determinannya tidak sama dengan nol, maka solusi akan ditampilkan di layar user.
public Matrix inverseWithIdentity()	Mengembalikan balikan matriks dengan menggunakan metode $AI = IA^{-1}$. Pada fungsi ini, pertama dibuat augmented matriks AI, yang merupakan gabungan antara matriks A dengan matriks identitas. Kemudian dilakukan eliminasi Gauss Jordan untuk mengubah menjadi dalam bentuk format IA^{-1} yang disimpan dalam variabel

	<p>IAinverse. Jika pada IAINverse tidak terbentuk matriks identitas atau tidak sesuai format IA^{-1} maka fungsi mengembalikan null, sedangkan jika sesuai format IA^{-1} maka fungsi akan mengembalikan balikan matriks dari matriks tersebut.</p>
<p>public Matrix inverseWithAdjMethod()</p>	<p>Mengembalikan balikan matriks dengan menggunakan metode adjoin matrix. Jika determinan dari matriks masukan adalah 0 maka fungsi akan mengembalikan nilai null. Jika determinan dari matriks tidak nol, maka fungsi akan mengembalikan balikan matriks.</p>
<p>public void polinomialInterpolation (Scanner scanner)</p>	<p>Prosedur akan meminta masukan dari user untuk menginterpolasi seberapa banyak titik koordinat dan nilai x yang hendak ditaksir oleh user. User tidak dapat memasukkan titik kurang dari dua. Kemudian, titik hasil input kemudian diolah menjadi sistem persamaan linier yang kemudian diubah menjadi augmented matrix dan disimpan ke dalam variabel dummyMatrix.</p> <p>Kemudian, dummyMatrix dioperasikan dengan metode Gauss, dan jika hasil Gauss tidak ada solusi, maka interpolasi tidak ada fungsi polinomial. Jika persamaan linier memiliki solusi, maka solusinya digunakan untuk membuat persamaan fungsi polinomial. Untuk penaksiran nilai, maka nilai inputan taksiran dimasukkan ke dalam fungsi polinomial $f(x)$. Maka, output dari prosedur ini ada dua, yaitu fungsi polinomial hasil interpolasi dan hasil taksiran dengan nilai inputan taksiran oleh user yang dimasukkan ke dalam fungsi polinomial.</p>
<p>public static void polynomialInterpolationByFile(String filename)</p>	<p>Prosedur akan membaca titik koordinat dan nilai x yang hendak ditaksir dari file <i>filename</i>. Kemudian, titik koordinat hasil</p>

	<p>bacaan kemudian diolah menjadi sistem persamaan linier yang kemudian diubah menjadi augmented matrix dan disimpan ke dalam variabel dummyMatrix.</p> <p>Kemudian, dummyMatrix dioperasikan dengan metode Gauss, dan jika hasil Gauss tidak ada solusi, maka interpolasi tidak ada fungsi polinomial. Jika persamaan linier memiliki solusi, maka solusinya digunakan untuk membuat persamaan fungsi polinomial. Untuk penaksiran nilai, maka nilai inputan taksiran dimasukkan ke dalam fungsi polinomial $f(x)$. Maka, output dari prosedur ini ada dua, yaitu fungsi polinomial hasil interpolasi dan hasil taksiran dengan nilai inputan taksiran oleh user yang dimasukkan ke dalam fungsi polinomial.</p>
public boolean isAllNull(Double [] row)	Mengembalikan true jika dalam array semuanya adalah null , dan mengembalikan false jika tidak semuanya null
public String outputSolution(Double [][] ans)	Mengembalikan sebuah string yang berupa solusi spl dalam bentuk ($x_1 = \dots, x_2 = \dots, x_3 = \dots$) dari array 2 dimensi yang merupakan solusi dari SPL dalam bentuk array 2 dimensi
public void OperationMatrix (Scanner scanner)	Procedure ini mengeluarkan solusi SPL dari matriks eselon atau matriks eselon tereduksi, apabila di baris terakhir semua kolom kecuali kolom terakhir bernilai nol, maka akan dikeluarkan output tidak ada solusi, jika matriks yang dilakukan operasi ada solusi tunggal , maka akan dilakukan print output solusi, kemudian jika solusi banyak , maka akan dikeluarkan output solusi SPL dalam bentuk variabel.
public Double[][] solutionOfGaussMethod()	Mengembalikan solution dalam bentuk array 2 dimensi yang dimana setiap baris mewakili satu solusi x ,

	dan setiap kolom mewakili banyak x sesuai dengan kolom indeksinya.
public void GaussMethod (Scanner scanner)	Prosedur ini mengubah matriks yang dilakukan operasi menjadi matriks eselon (leading one), kemudian memanggil prosedur OperationMatrix() untuk mengeluarkan output solusi SPL dari matriks eselon tersebut.
public void GaussJordanMethod (Scanner scanner)	Prosedur ini mengubah matriks yang dilakukan operasi menjadi matriks eselon tereduksi, kemudian memanggil prosedur OperationMatrix() untuk mengeluarkan output solusi SPL dari matriks eselon tereduksi tersebut.
public static void MultipleLinearRegression (Scanner scanner)	Prosedur ini meminta user memilih untuk menginput matriks dari input dari terminal atau file, jika memilih input terminal, maka akan diminta memasukkan jumlah variabel dan sampel, lalu dilanjutkan untuk menginput sampel. Jika memilih untuk menginput matriks dari file, maka akan diminta nama file. Kemudian akan dibuat normal equation dari matriks yang diinput, setelah itu dicari solusi dari normal equation dengan menggunakan metode gauss. Kemudian program akan meminta nilai-nilai x yang akan ditaksir, setelah user menginput nilai-nilai x tersebut, kemudian dilakukan perhitungan untuk mendapatkan nilai tafsiran.
public static void OutputToFile(Scanner scanner, String output)	Prosedur ini meminta user menginput nama file yang akan dituliskan output, setelah user menginput nama file, akan dilakukan pembuatan file dengan nama yang diinput, lalu akan dilakukan penulisan string pada file yang ditulis
public String MatrixToString()	Fungsi ini mengembalikan string yang merupakan sebuah matriks

	yang telah dikonversi menjadi string, yang kemudian akan digunakan untuk melakukan output di file
public static Matrix getmTurunanfile(String filename)	Fungsi ini menerima string berisi nama file berisi matriks turunan berukuran 4x4, lalu dibaca dan mengembalikan matriks hasil bacaan.
public static Matrix bacamTurunanTerminal(Scanner scanner)	Fungsi ini meminta user untuk menginput matriks turunan berukuran 4x4 dari terminal, lalu mengembalikan matriks hasil inputan.
public Matrix transformKeMatriksTurunan(Matrix mInput)	Fungsi ini membaca matriks input yang dipastikan berukuran 4x4 dan mengubah matriks tersebut ke bentuk 16x1 dan mengembalikan matriks hasil perubahan.
public static int checkXBicubic(int row)	Fungsi ini membaca angka input yang merepresentasikan baris matriks dan mengembalikan 1 jika $row \equiv 0 \pmod{4}$ atau $row \equiv 2 \pmod{4}$ dan mengembalikan 0 untuk kasus sisa. Fungsi ini bertujuan untuk menentukan nilai x yang dipakai dalam pembuatan matriks X.
public static int checkYBicubic(int row)	Fungsi ini membaca angka input yang merepresentasikan baris matriks dan mengembalikan 1 jika $row \equiv 0 \pmod{4}$ atau $row \equiv 1 \pmod{4}$ dan mengembalikan 0 untuk kasus sisa. Fungsi ini bertujuan untuk menentukan nilai y yang dipakai dalam pembuatan matriks X.
public static Matrix makeMatriksX()	Fungsi ini mengembalikan matriks X berukuran 16x16 yang berisi koefisien matriksTurunan untuk bicubic Spline Interpolation. makeMatriksX memanggil fungsi checkXBicubic(int row) dan fungsi checkYBicubic(int row) untuk mengecek nilai x dan y yang akan

	dipakai. Row nol-tiga berupa koefisien untuk variabel yang tidak diturunkan, row empat-tujuh berupa koefisien untuk variabel yang diturunkan terhadap x, row delapan-sebelah untuk variabel yang diturunkan terhadap y, dan row dua belas-lima belas untuk variabel yang diturunkan terhadap x dan y.
<pre>public static double[] listAandB(String filename)</pre>	Fungsi ini menerima string berisi nama file yang akan diambil nilai a dan b nya yang berada di baris ke 5 dan dimasukkan ke dalam double[] dan mengembalikan array tersebut.
<pre>public static void bicubicSplineInterpolation()</pre>	<p>prosedur ini bertujuan untuk mengeluarkan nilai Interpolasi bicubic spline ke layar. Prosedur ini memberikan dua pilihan kepada user untuk menginput matriks dan koefisien interpolasi (a dan b) lewat terminal atau lewat file txt.</p> <p>Jika memilih terminal, maka dipanggil fungsi <code>bacamTurunanTerminal</code> yang akan ditransform menggunakan fungsi <code>transformKeMatriksTurunan</code> dan meminta user untuk menginput nilai a dan b dengan range [0..1]. Jika memilih file, maka user diminta memasukkan nama file lalu dipanggil fungsi <code>getmTurunanFile</code> yang akan ditransform menggunakan fungsi <code>transformKeMatriksTurunan</code> dan <code>listAandB</code>.</p> <p>Kemudian dibuat matriksX dengan memanggil fungsi <code>makeMatriksX</code> lalu mengalikan matriksX dengan matriksTurunan untuk menghasilkan <code>matrixKali</code>. Lalu menggunakan <code>matrixKali</code> serta nilai a dan b, didapatkan hasil interpolasi di titik (a,b) yang dimunculkan ke layar.</p>

2. Main.java

- Fungsi / Prosedur

Fungsi / Prosedur	Deskripsi
public static void SHOW_GREETINGS_TO_USER()	Menampilkan pesan selamat datang kepada user
public static void CLEAR_TERMINAL()	Melakukan pembersihan pada terminal
public static Matrix INPUT_MATRIX_SUBMENU(Scanner scanner, int sq)	Menampilkan pesan pilihan untuk jenis pembacaan matriks, bisa input dengan terminal maupun membaca dari file .txt. Kemudian, meminta masukan dari user dengan memanggil fungsi ;GET_VALID_CHOICE(scanner, 1, 2, 5, 1) dan masukan matriks dengan memanggil fungsi INPUT_MATRIX(choice, scanner, sq)
public static Matrix INPUT_MATRIX(int choice, Scanner scanner, int sq)	Jika choice bernilai 1 dan sq bernilai 0, maka program akan meminta user untuk memasukkan banyak baris, banyak kolom, dan elemen-elemen dari matriks yang hendak diinisialisasi. Jika choice bernilai 1 dan sq bernilai 1 (yang berarti masukan pasti merupakan matriks persegi, maka program akan meminta user untuk input N yang merupakan banyak baris dan kolom matriks, serta user mengisi elemen-elemen dari matriks yang diinisialisasi. Jika choice bernilai 2, maka pembacaan matriks akan dilakukan melalui pembacaan dari file .txt. Pada akhirnya, fungsi ini akan mengembalikan matriks hasil inputan dari user ataupun bacaan dari file .txt.
public static void SHOW_MAIN_MENU()	Menampilkan pesan berupa pilihan yang dapat dipilih dari menu utama program
public static void SHOW_DETERMINANT_SUBMENU()	Menampilkan pesan berupa pilihan yang dapat dipilih dari submenu determinan
public static void	Menampilkan pesan berupa pilihan

SHOW_LINEAR_EQUATION_SUBMENU()	yang dapat dipilih dari submenu solusi sistem persamaan linear
public static void SHOW_INVERSE_SUBMENU()	Menampilkan pesan berupa pilihan yang dapat dipilih dari submenu matriks balikan
public static void WRONG_INPUT_REMINDER()	Menampilkan pesan pengingat kepada user untuk melakukan inputan yang sesuai
public static void ASK_FOR_CHOICE_MESSAGE()	Menampilkan pesan untuk bertanya pilihan apa yang hendak dipilih oleh user
public static boolean REUSE_CONFIRMATION(Scanner scanner)	Fungsi ini bertujuan untuk melakukan konfirmasi kepada user apakah masih ingin melanjutkan penggunaan program setelah sebuah operasi telah selesai dilakukan. Jika fungsi ini mengeluarkan true, maka program akan berjalan ulang kembali.
public static int GET_VALID_CHOICE(Scanner scanner, int leastChoice, int highestChoice, int menu, int sq)	Fungsi ini bertujuan untuk memastikan bahwa inputan user telah benar sesuai dengan batas yang ditentukan yaitu $\text{leastChoice} \leq (\text{choice}) \leq \text{highestChoice}$. Jika berada di luar range tersebut, maka akan diminta kembali inputan dari user. Jika inputan user valid, maka akan ditampilkan pesan menu sesuai dengan parameter menu yang digunakan. Parameter sq hanya digunakan ketika penerimaan inputan SPL, determinan, dan invers untuk mengontrol jenis inputan berupa matriks augmented atau matriks $N \times N$.

BAB 4

EKSPERIMEN

4.1 Temukan solusi SPL $Ax = b$

a.
$$A = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 2 & 5 & -7 & -5 \\ 2 & -1 & 1 & 3 \\ 5 & 2 & -4 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ -2 \\ 4 \\ 6 \end{bmatrix}$$

Metode Eliminasi Gauss

Input Terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 1 -1 -1 1
2 5 -7 -5 -2
2 -1 1 3 4
5 2 -4 2 6

Here is the matrix read by the program:
1.000000    1.000000   -1.000000   -1.000000    1.000000
2.000000    5.000000   -7.000000   -5.000000   -2.000000
2.000000   -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000   -4.000000    2.000000    6.000000

The result of Gauss method for the system of linear equation is the following :
```

This is the the OBE result:				
1.000000	1.000000	-1.000000	-1.000000	1.000000
0.000000	1.000000	-1.666667	-1.000000	-1.333333
0.000000	0.000000	1.000000	-1.000000	1.000000
0.000000	0.000000	0.000000	0.000000	1.000000

```
There are no solution.
```

Input dari file

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-a.txt

Here is the matrix read by the program:
1.000000    1.000000   -1.000000   -1.000000    1.000000
2.000000    5.000000   -7.000000   -5.000000   -2.000000
2.000000   -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000   -4.000000    2.000000    6.000000

The result of Gauss method for the system of linear equation is the following :
```

This is the the OBE result:				
1.000000	1.000000	-1.000000	-1.000000	1.000000
0.000000	1.000000	-1.666667	-1.000000	-1.333333
0.000000	0.000000	1.000000	-1.000000	1.000000
0.000000	0.000000	0.000000	0.000000	1.000000

```
There are no solution.
```

Metode Eliminasi Gauss-Jordan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 1 -1 -1 1
2 5 -7 -5 -2
2 -1 1 3 4
5 2 -4 2 6

Here is the matrix read by the program:
1.000000    1.000000    -1.000000    -1.000000    1.000000
2.000000    5.000000    -7.000000    -5.000000    -2.000000
2.000000    -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000    -4.000000    2.000000    6.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.666667    0.000000
0.000000    1.000000    0.000000    -2.666667    0.000000
0.000000    0.000000    1.000000    -1.000000    0.000000
0.000000    0.000000    0.000000    0.000000    1.000000

There are no solution.

```

input dari file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-a.txt

Here is the matrix read by the program:
1.000000    1.000000    -1.000000    -1.000000    1.000000
2.000000    -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000    -4.000000    2.000000    6.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.666667    0.000000
0.000000    1.000000    0.000000    -2.666667    0.000000
0.000000    0.000000    1.000000    -1.000000    0.000000
0.000000    0.000000    0.000000    0.000000    1.000000

There are no solution.

```


Metode Matriks Balikan

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 1 -1 -1 1
2 5 -7 -5 -2
2 -1 1 3 4
5 2 -4 2 6

Here is the matrix read by the program:
1.000000    1.000000   -1.000000   -1.000000    1.000000
2.000000    5.000000   -7.000000   -5.000000   -2.000000
2.000000   -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000   -4.000000    2.000000    6.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.
```

input dari file

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-a.txt

Here is the matrix read by the program:
1.000000    1.000000   -1.000000    1.000000
2.000000    5.000000   -7.000000   -5.000000   -2.000000
2.000000   -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000   -4.000000    2.000000    6.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.
```

Metode Cramer

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 1 -1 -1 1
2 5 -7 -5 -2
2 -1 1 3 4
5 2 -4 2 6

Here is the matrix read by the program:
1.000000    1.000000   -1.000000    1.000000
2.000000    5.000000   -7.000000   -5.000000   -2.000000
2.000000   -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000   -4.000000    2.000000    6.000000

The result of Cramer method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.
```

input dari file

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: dadawdaf.txt
File not found in test folder. Make sure your file is existed in that folder.
Input the filename and don't forget to include .txt: testcase1-a.txt

Here is the matrix read by the program:
1.000000    1.000000    -1.000000    -1.000000    1.000000
2.000000    5.000000    -7.000000    -5.000000    -2.000000
2.000000    -1.000000    1.000000    3.000000    4.000000
5.000000    2.000000    -4.000000    2.000000    6.000000

The result of Cramer method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.
```

b.

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 0 & -3 & 0 \\ 2 & -1 & 0 & 1 & -1 \\ -1 & 2 & 0 & -2 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 6 \\ 5 \\ -1 \end{bmatrix}$$

Metode Gauss

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 6
Enter your matrix with 4x6 matrix size.
1 -1 0 0 1 3
1 1 0 -3 0 6
2 -1 0 1 -1 5
-1 2 0 -2 -1 -1

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000    -3.000000    0.000000    6.000000
2.000000    -1.000000    0.000000    1.000000    -1.000000    5.000000
-1.000000    2.000000    0.000000    -2.000000    -1.000000    -1.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
0.000000    1.000000    0.000000    -1.500000    -0.500000    1.500000
0.000000    0.000000    0.000000    1.000000    -1.000000    -1.000000
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 1.0000e + 3.0000
x2 = 2.0000e
x3 = c
x4 = 1.0000e - 1.0000
x5 = e
```

input dari file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-b.txt

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000    -3.000000    0.000000    6.000000
2.000000    -1.000000    0.000000    1.000000    -1.000000    5.000000
-1.000000    2.000000    0.000000    -2.000000    -1.000000    -1.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
0.000000    1.000000    0.000000    -1.500000    -0.500000    1.500000
0.000000    0.000000    0.000000    1.000000    -1.000000    -1.000000
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 1.0000e + 3.0000
x2 = 2.0000e
x3 = c
x4 = 1.0000e - 1.0000
x5 = e

```

Metode Gauss-Jordan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 6
Enter your matrix with 4x6 matrix size.
1 -1 0 0 1 3
1 1 0 -3 0 6
2 -1 0 1 -1 5
-1 2 0 -2 -1 -1

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000    -3.000000    0.000000    6.000000
2.000000    -1.000000    0.000000    1.000000    -1.000000    5.000000
-1.000000    2.000000    0.000000    -2.000000    -1.000000    -1.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000    -1.000000    3.000000
0.000000    1.000000    0.000000    0.000000    -2.000000    0.000000
0.000000    0.000000    0.000000    1.000000    -1.000000    -1.000000
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 1.0000e + 3.0000
x2 = 2.0000e
x3 = c
x4 = 1.0000e - 1.0000
x5 = e

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-b.txt

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000    -3.000000    0.000000    6.000000
2.000000    -1.000000    0.000000    1.000000    -1.000000    5.000000
-1.000000    2.000000    0.000000    -2.000000    -1.000000    -1.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000    -1.000000    3.000000
0.000000    1.000000    0.000000    0.000000    -2.000000    0.000000
0.000000    0.000000    0.000000    1.000000    -1.000000    -1.000000
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 1.0000e + 3.0000
x2 = 2.0000e
x3 = c
x4 = 1.0000e - 1.0000
x5 = e

```

Metode Matriks Balikan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 6
Enter your matrix with 4x6 matrix size.
1 -1 0 1 3
1 1 0 -3 0 6
2 -1 0 1 -1 5
-1 2 0 -2 -1 -1

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000    -3.000000    0.000000    6.000000
2.000000    -1.000000    0.000000    1.000000    -1.000000    5.000000
-1.000000    2.000000    0.000000    -2.000000    -1.000000    -1.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-b.txt

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000    -3.000000    0.000000    6.000000
2.000000    -1.000000    0.000000    1.000000    -1.000000    5.000000
-1.000000    2.000000    0.000000    -2.000000    -1.000000    -1.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.

```

Metode Cramer

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 6
Enter your matrix with 4x6 matrix size.
1 -1 0 0 1 3
1 1 0 -3 0 6
2 -1 0 1 -1 5
-1 2 0 -2 -1 -1

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000   -3.000000    0.000000    6.000000
2.000000   -1.000000    0.000000    1.000000   -1.000000    5.000000
-1.000000    2.000000    0.000000   -2.000000   -1.000000   -1.000000

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.

```

input dari file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-b.txt

Here is the matrix read by the program:
1.000000    -1.000000    0.000000    0.000000    1.000000    3.000000
1.000000    1.000000    0.000000   -3.000000    0.000000    6.000000
2.000000   -1.000000    0.000000    1.000000   -1.000000    5.000000
-1.000000    2.000000    0.000000   -2.000000   -1.000000   -1.000000

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.

```

c.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$$

Metode Gauss

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 3
Input how many columns the matrix will have (enter a positive integer): 7
Enter your matrix with 3x7 matrix size.
0 1 0 0 1 0 2
0 0 0 1 1 0 -1
0 1 0 0 0 1 1

Here is the matrix read by the program:
0.000000      1.000000      0.000000      0.000000      1.000000      0.000000      2.000000
0.000000      0.000000      0.000000      1.000000      1.000000      0.000000      -1.000000
0.000000      1.000000      0.000000      0.000000      0.000000      1.000000      1.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
0.000000      1.000000      0.000000      0.000000      1.000000      0.000000      2.000000
0.000000      0.000000      0.000000      1.000000      1.000000      0.000000      -1.000000
0.000000      0.000000      0.000000      0.000000      1.000000      -1.000000      1.000000

x1 = a
x2 = -1.0000f + 1.0000
x3 = c
x4 = -1.0000f - 2.0000
x5 = 1.0000f + 1.0000
x6 = f

```

input dari file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-c.txt

Here is the matrix read by the program:
0.000000      1.000000      0.000000      0.000000      1.000000      0.000000      2.000000
0.000000      0.000000      0.000000      1.000000      1.000000      0.000000      -1.000000
0.000000      1.000000      0.000000      0.000000      0.000000      1.000000      1.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
0.000000      1.000000      0.000000      0.000000      1.000000      0.000000      2.000000
0.000000      0.000000      0.000000      1.000000      1.000000      0.000000      -1.000000
0.000000      0.000000      0.000000      0.000000      1.000000      -1.000000      1.000000

x1 = a
x2 = -1.0000f + 1.0000
x3 = c
x4 = -1.0000f - 2.0000
x5 = 1.0000f + 1.0000
x6 = f

```

Metode Gauss-Jordan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 3
Input how many columns the matrix will have (enter a positive integer): 7
Enter your matrix with 3x7 matrix size.
0 1 0 0 1 0 2
0 0 0 1 1 0 -1
0 1 0 0 0 1 1

Here is the matrix read by the program:
0.000000    1.000000    0.000000    0.000000    1.000000    0.000000    2.000000
0.000000    0.000000    0.000000    1.000000    1.000000    0.000000    -1.000000
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000
0.000000    0.000000    0.000000    1.000000    0.000000    1.000000    -2.000000
0.000000    0.000000    0.000000    0.000000    1.000000    -1.000000    1.000000

x1 = a
x2 = -1.0000f + 1.0000
x3 = c
x4 = -1.0000f - 2.0000
x5 = 1.0000f + 1.0000
x6 = f

```

input dari file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-c.txt

Here is the matrix read by the program:
0.000000    1.000000    0.000000    0.000000    1.000000    0.000000    2.000000
0.000000    0.000000    0.000000    1.000000    1.000000    0.000000    -1.000000
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000
0.000000    0.000000    0.000000    1.000000    0.000000    1.000000    -2.000000
0.000000    0.000000    0.000000    0.000000    1.000000    -1.000000    1.000000

x1 = a
x2 = -1.0000f + 1.0000
x3 = c
x4 = -1.0000f - 2.0000
x5 = 1.0000f + 1.0000
x6 = f

```

Metode Matriks Balikan

input dari terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 3
Input how many columns the matrix will have (enter a positive integer): 7
Enter your matrix with 3x7 matrix size.
0 1 0 0 1 0 2
0 0 0 1 1 0 -1
0 1 0 0 0 1 1

Here is the matrix read by the program:
0.000000    1.000000    0.000000    0.000000    1.000000    0.000000    2.000000
0.000000    0.000000    0.000000    1.000000    1.000000    0.000000    -1.000000
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.
```

input dari file

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-c.txt

Here is the matrix read by the program:
0.000000    1.000000    0.000000    0.000000    1.000000    0.000000    2.000000
0.000000    0.000000    0.000000    1.000000    1.000000    0.000000    -1.000000
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.
```

Metode Cramer

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 3
Input how many columns the matrix will have (enter a positive integer): 7
Enter your matrix with 3x7 matrix size.
0 1 0 0 1 0 2
0 0 0 1 1 0 -1
0 1 0 0 0 1 1

Here is the matrix read by the program:
0.000000    1.000000    0.000000    0.000000    1.000000    0.000000    2.000000
0.000000    0.000000    0.000000    1.000000    1.000000    0.000000    -1.000000
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.
```


input dari file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-c.txt

Here is the matrix read by the program:
0.000000    1.000000    0.000000    0.000000    1.000000    0.000000    2.000000
0.000000    0.000000    0.000000    1.000000    1.000000    0.000000    -1.000000
0.000000    1.000000    0.000000    0.000000    0.000000    1.000000    1.000000

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.

```

d.

$$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \dots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \dots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \dots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \dots & \frac{1}{2n+1} \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

H adalah matriks *Hilbert*. Cobakan untuk $n = 6$ dan $n = 10$.

Untuk $n = 6$

Metode gauss

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-1.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    1.000000
0.000000    1.000000    1.000600    0.900360    0.799520    0.715486    -6.002401
0.000000    0.000000    1.000000    1.493352    1.741950    1.784849    30.302612
0.000000    0.000000    0.000000    1.000000    1.200900    1.886398    -84.447194
0.000000    0.000000    0.000000    0.000000    1.000000    1.987051    -110.963171
0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    -184.223716

x1 = 7.6386
x2 = -18.5442
x3 = -20.6288
x4 = -43.2759
x5 = 255.0987
x6 = -184.2237

```

Metode Gauss Jordan

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-1.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000    0.000000    0.000000    7.638612
0.000000    1.000000    0.000000    0.000000    0.000000    0.000000    -18.544237
0.000000    0.000000    1.000000    0.000000    0.000000    0.000000    -20.628838
0.000000    0.000000    0.000000    1.000000    0.000000    0.000000    -43.275878
0.000000    0.000000    0.000000    0.000000    1.000000    0.000000    255.098692
0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    -184.223716

x1 = 7.6386
x2 = -18.5442
x3 = -20.6288
x4 = -43.2759
x5 = 255.0987
x6 = -184.2237

```

Metode Matriks Balikan

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-1.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
X1 bernilai 7.6386
X2 bernilai -18.5442
X3 bernilai -20.6288
X4 bernilai -43.2759
X5 bernilai 255.0987
X6 bernilai -184.2237

```

Metode Cramer

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-1.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000

The result of Cramer method for the system of linear equation is the following :
X0 bernilai 7.6386
X1 bernilai -18.5442
X2 bernilai -20.6288
X3 bernilai -43.2759
X4 bernilai 255.0987
X5 bernilai -184.2237
```

n = 10

Metode gauss

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-2.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.000000
0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.000000
0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.000000
0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.000000
0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.052600    0.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    1.000000
0.000000    1.000000    1.000600    0.900360    0.799520    0.715486    0.642857    0.583433    0.533613    0.490996    -6.002401
0.000000    0.000000    1.000000    1.493352    1.741950    1.784849    1.794437    1.761602    1.704510    1.641321    30.302612
0.000000    0.000000    0.000000    1.000000    1.200900    1.886398    2.218058    2.393127    2.532748    2.689463    -84.447194
0.000000    0.000000    0.000000    0.000000    1.000000    1.987051    2.516604    2.967838    3.481581    3.672839    -110.963171
0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    1.318414    1.335651    1.713628    1.978324    -184.223716
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    1.206176    1.193879    3.000267    40.796414
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    -222.195702    -46.017181    -29529.824312
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    0.210014    132.774611
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    1.406298

x1 = 9.7879
x2 = -40.3226
x3 = -5.8483
x4 = 35.5465
x5 = 245.3990
x6 = -261.0552
x7 = -86.8636
x8 = -28.7864
x9 = 132.4793
x10 = 1.4063
```

Metode Gauss Jordan

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-2.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.000000
0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.000000
0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.000000
0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.000000
0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.052600    0.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    9.787929
0.000000    1.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    -40.322564
0.000000    0.000000    1.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    -5.848280
0.000000    0.000000    0.000000    1.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    35.546459
0.000000    0.000000    0.000000    0.000000    1.000000    0.000000    0.000000    0.000000    0.000000    0.000000    245.398972
0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    0.000000    0.000000    0.000000    0.000000    -261.055165
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    0.000000    0.000000    0.000000    -86.863613
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    0.000000    0.000000    -28.786397
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    0.000000    132.479268
0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    0.000000    1.000000    1.406298

x1 = 9.7879
x2 = -40.3226
x3 = -5.8483
x4 = 35.5465
x5 = 245.3990
x6 = -261.0552
x7 = -86.8636
x8 = -28.7864
x9 = 132.4793
x10 = 1.4063

```

Metode Matriks balikan

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-2.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.000000
0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.000000
0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.000000
0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.000000
0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.052600    0.000000

The result of Inverse Method (AX=B -> X=(A^-1)B) method for the system of linear equation is the following :
X1 bernilai 9.7879
X2 bernilai -40.3226
X3 bernilai -5.8483
X4 bernilai 35.5465
X5 bernilai 245.3990
X6 bernilai -261.0552
X7 bernilai -86.8636
X8 bernilai -28.7864
X9 bernilai 132.4793
X10 bernilai 1.4063

```

Metode Cramer

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase1-d-2.txt

Here is the matrix read by the program:
1.000000    0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    1.000000
0.500000    0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.000000
0.333300    0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.000000
0.250000    0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.000000
0.200000    0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.000000
0.166600    0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.000000
0.142900    0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.000000
0.125000    0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.000000
0.111100    0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.000000
0.100000    0.090900    0.083300    0.076900    0.071400    0.066600    0.062500    0.058800    0.055500    0.052600    0.000000

The result of Cramer method for the system of linear equation is the following :
X0 bernilai 9.7879
X1 bernilai -40.3226
X2 bernilai -5.8483
X3 bernilai 35.5465
X4 bernilai 245.3990
X5 bernilai -261.0552
X6 bernilai -86.8636
X7 bernilai -28.7864
X8 bernilai 132.4793
X9 bernilai 1.4063

```

4.2 SPL berbentuk matriks augmented

a.

$$\begin{bmatrix} 1 & -1 & 2 & -1 & -1 \\ 2 & 1 & -2 & -2 & -2 \\ -1 & 2 & -4 & 1 & 1 \\ 3 & 0 & 0 & -3 & -3 \end{bmatrix}.$$

Metode Eliminasi Gauss

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 -1 2 -1 -1
2 1 -2 -2 -2
-1 2 -4 1 1
3 0 0 -3 -3

Here is the matrix read by the program:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
2.000000    1.000000    -2.000000    -2.000000    -2.000000
-1.000000    2.000000    -4.000000    1.000000    1.000000
3.000000    0.000000    0.000000    -3.000000    -3.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
0.000000    1.000000    -2.000000    0.000000    0.000000
0.000000    0.000000    0.000000    0.000000    0.000000
0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 1.0000d - 1.0000
x2 = 2.0000c
x3 = c
x4 = d
```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-a.txt

Here is the matrix read by the program:
1.000000      -1.000000      2.000000      -1.000000      -1.000000
2.000000      1.000000      -2.000000      -2.000000      -2.000000
-1.000000      2.000000      -4.000000      1.000000      1.000000
3.000000      0.000000      0.000000      -3.000000      -3.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000      -1.000000      2.000000      -1.000000      -1.000000
0.000000      1.000000      -2.000000      0.000000      0.000000
0.000000      0.000000      0.000000      0.000000      0.000000
0.000000      0.000000      0.000000      0.000000      0.000000

x1 = 1.0000d - 1.0000
x2 = 2.0000c
x3 = c
x4 = d

```

Metode Gauss-Jordan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 -1 2 -1 -1
2 1 -2 -2 -2
-1 2 -4 1 1
3 0 0 -3 -3

Here is the matrix read by the program:
1.000000      -1.000000      2.000000      -1.000000      -1.000000
2.000000      1.000000      -2.000000      -2.000000      -2.000000
-1.000000      2.000000      -4.000000      1.000000      1.000000
3.000000      0.000000      0.000000      -3.000000      -3.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000      0.000000      0.000000      -1.000000      -1.000000
0.000000      1.000000      -2.000000      0.000000      0.000000
0.000000      0.000000      0.000000      0.000000      0.000000
0.000000      0.000000      0.000000      0.000000      0.000000

x1 = 1.0000d - 1.0000
x2 = 2.0000c
x3 = c
x4 = d

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-a.txt

Here is the matrix read by the program:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
2.000000     1.000000    -2.000000    -2.000000    -2.000000
-1.000000     2.000000    -4.000000     1.000000     1.000000
3.000000     0.000000     0.000000    -3.000000    -3.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000     0.000000     0.000000    -1.000000    -1.000000
0.000000     1.000000    -2.000000     0.000000     0.000000
0.000000     0.000000     0.000000     0.000000     0.000000
0.000000     0.000000     0.000000     0.000000     0.000000

x1 = 1.0000d - 1.0000
x2 = 2.0000c
x3 = c
x4 = d

```

Metode Matriks Balikan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 -1 2 -1 -1
2 1 -2 -2 -2
-1 2 -4 1 1
3 0 0 -3 -3

Here is the matrix read by the program:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
2.000000     1.000000    -2.000000    -2.000000    -2.000000
-1.000000     2.000000    -4.000000     1.000000     1.000000
3.000000     0.000000     0.000000    -3.000000    -3.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-a.txt

Here is the matrix read by the program:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
2.000000     1.000000    -2.000000    -2.000000    -2.000000
-1.000000     2.000000    -4.000000     1.000000     1.000000
3.000000     0.000000     0.000000    -3.000000    -3.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.

```


Metode Cramer

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
1 -1 2 -1 -1
2 1 -2 -2 -2
-1 2 -4 1 1
3 0 0 -3 -3

Here is the matrix read by the program:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
2.000000     1.000000   -2.000000   -2.000000   -2.000000
-1.000000    2.000000   -4.000000    1.000000    1.000000
3.000000     0.000000    0.000000   -3.000000   -3.000000

The result of Cramer method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed?
2
Input the filename and don't forget to include .txt: testcase2-a.txt

Here is the matrix read by the program:
1.000000    -1.000000    2.000000    -1.000000    -1.000000
2.000000     1.000000   -2.000000   -2.000000   -2.000000
-1.000000    2.000000   -4.000000    1.000000    1.000000
3.000000     0.000000    0.000000   -3.000000   -3.000000

The result of Cramer method for the system of linear equation is the following :
This SPL does not have any solution because the matrix determinant is 0.

```

b.

$$\begin{bmatrix} 2 & 0 & 8 & 0 & 8 \\ 0 & 1 & 0 & 4 & 6 \\ -4 & 0 & 6 & 0 & 6 \\ 0 & -2 & 0 & 3 & -1 \\ 2 & 0 & -4 & 0 & -4 \\ 0 & 1 & 0 & -2 & 0 \end{bmatrix}.$$

Metode Gauss

input terminal

```

1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 6
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 6x5 matrix size.
2 0 8 0 8
0 1 0 4 6
-4 0 6 0 6
0 -2 0 3 -1
2 0 -4 0 -4
0 1 0 -2 0

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    4.000000    0.000000    4.000000
0.000000    1.000000    0.000000    4.000000    6.000000
0.000000    0.000000    1.000000    0.000000    1.000000
0.000000    0.000000    0.000000    1.000000    1.000000
0.000000    0.000000    0.000000    0.000000    0.000000
0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 0.0000
x2 = 2.0000
x3 = 1.0000
x4 = 1.0000

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-b.txt

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    4.000000    0.000000    4.000000
0.000000    1.000000    0.000000    4.000000    6.000000
0.000000    0.000000    1.000000    0.000000    1.000000
0.000000    0.000000    0.000000    1.000000    1.000000
0.000000    0.000000    0.000000    0.000000    0.000000
0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 0.0000
x2 = 2.0000
x3 = 1.0000
x4 = 1.0000

```

Metode Gauss-Jordan

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 6
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 6x5 matrix size.
2 0 8 0 8
0 1 0 4 6
-4 0 6 0 6
0 -2 0 3 -1
2 0 -4 0 -4
0 1 0 -2 0

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000    0.000000
0.000000    1.000000    0.000000    0.000000    2.000000
0.000000    0.000000    1.000000    0.000000    1.000000
0.000000    0.000000    0.000000    1.000000    1.000000
0.000000    0.000000    0.000000    0.000000    0.000000
0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 0.0000
x2 = 2.0000
x3 = 1.0000
x4 = 1.0000
```

input file

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-b.txt

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000    0.000000
0.000000    1.000000    0.000000    0.000000    2.000000
0.000000    0.000000    1.000000    0.000000    1.000000
0.000000    0.000000    0.000000    1.000000    1.000000
0.000000    0.000000    0.000000    0.000000    0.000000
0.000000    0.000000    0.000000    0.000000    0.000000

x1 = 0.0000
x2 = 2.0000
x3 = 1.0000
x4 = 1.0000
```

Metode Matriks Balikan

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 6
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 6x5 matrix size.
2 0 8 0 8
0 1 0 4 6
-4 0 6 0 6
0 -2 0 3 -1
2 0 -4 0 -4
0 1 0 -2 0

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.
```

input file

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-b.txt

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.
```

Metode Cramer

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 6
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 6x5 matrix size.
2 0 8 0 8
0 1 0 4 6
-4 0 6 0 6
0 -2 0 3 -1
2 0 -4 0 -4
0 1 0 -2 0

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.
```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase2-b.txt

Here is the matrix read by the program:
2.000000    0.000000    8.000000    0.000000    8.000000
0.000000    1.000000    0.000000    4.000000    6.000000
-4.000000    0.000000    6.000000    0.000000    6.000000
0.000000    -2.000000    0.000000    3.000000    -1.000000
2.000000    0.000000    -4.000000    0.000000    -4.000000
0.000000    1.000000    0.000000    -2.000000    0.000000

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.

```

4.3 SPL Berbentuk

a.

$$\begin{aligned}
 8x_1 + x_2 + 3x_3 + 2x_4 &= 0 \\
 2x_1 + 9x_2 - x_3 - 2x_4 &= 1 \\
 x_1 + 3x_2 + 2x_3 - x_4 &= 2 \\
 x_1 + \quad \quad 6x_3 + 4x_4 &= 3
 \end{aligned}$$

Metode Gauss

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
8 1 3 2 0
2 9 -1 -2 1
1 3 2 -1 2
1 0 6 4 3

Here is the matrix read by the program:
8.000000    1.000000    3.000000    2.000000    0.000000
2.000000    9.000000    -1.000000    -2.000000    1.000000
1.000000    3.000000    2.000000    -1.000000    2.000000
1.000000    0.000000    6.000000    4.000000    3.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.125000    0.375000    0.250000    0.000000
0.000000    1.000000    -0.200000    -0.285714    0.114286
0.000000    0.000000    1.000000    -0.194805    0.759740
0.000000    0.000000    0.000000    1.000000    -0.258108

x1 = -0.2243
x2 = 0.1824
x3 = 0.7095
x4 = -0.2581

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase3-a.txt

Here is the matrix read by the program:
8.000000    1.000000    3.000000    2.000000    0.000000
2.000000    9.000000   -1.000000   -2.000000    1.000000
1.000000    3.000000    2.000000   -1.000000    2.000000
1.000000    0.000000    6.000000    4.000000    3.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.125000    0.375000    0.250000    0.000000
0.000000    1.000000   -0.200000   -0.285714    0.114286
0.000000    0.000000    1.000000   -0.194805    0.759740
0.000000    0.000000    0.000000    1.000000   -0.258108

x1 = -0.2243
x2 = 0.1824
x3 = 0.7095
x4 = -0.2581

```

Metode Gauss-Jordan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
8 1 3 2 0
2 9 -1 -2 1
1 3 2 -1 2
1 0 6 4 3

Here is the matrix read by the program:
8.000000    1.000000    3.000000    2.000000    0.000000
2.000000    9.000000   -1.000000   -2.000000    1.000000
1.000000    3.000000    2.000000   -1.000000    2.000000
1.000000    0.000000    6.000000    4.000000    3.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000    0.000000    0.000000    0.000000   -0.224324
0.000000    1.000000    0.000000    0.000000    0.182432
0.000000    0.000000    1.000000    0.000000    0.709459
0.000000    0.000000    0.000000    1.000000   -0.258108

x1 = -0.2243
x2 = 0.1824
x3 = 0.7095
x4 = -0.2581

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase3-a.txt

Here is the matrix read by the program:
8.000000      1.000000      3.000000      2.000000      0.000000
2.000000      9.000000     -1.000000     -2.000000      1.000000
1.000000      3.000000      2.000000     -1.000000      2.000000
1.000000      0.000000      6.000000      4.000000      3.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000      0.000000      0.000000      0.000000     -0.224324
0.000000      1.000000      0.000000      0.000000      0.182432
0.000000      0.000000      1.000000      0.000000      0.709459
0.000000      0.000000      0.000000      1.000000     -0.258108

x1 = -0.2243
x2 = 0.1824
x3 = 0.7095
x4 = -0.2581

```

Metode Matriks Balikan

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
8 1 3 2 0
2 9 -1 -2 1
1 3 2 -1 2
1 0 6 4 3

Here is the matrix read by the program:
8.000000      1.000000      3.000000      2.000000      0.000000
2.000000      9.000000     -1.000000     -2.000000      1.000000
1.000000      3.000000      2.000000     -1.000000      2.000000
1.000000      0.000000      6.000000      4.000000      3.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
X1 bernilai -0.2243
X2 bernilai 0.1824
X3 bernilai 0.7095
X4 bernilai -0.2581

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase3-a.txt

Here is the matrix read by the program:
8.000000      1.000000      3.000000      2.000000      0.000000
2.000000      9.000000     -1.000000     -2.000000      1.000000
1.000000      3.000000      2.000000     -1.000000      2.000000
1.000000      0.000000      6.000000      4.000000      3.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
X1 bernilai -0.2243
X2 bernilai 0.1824
X3 bernilai 0.7095
X4 bernilai -0.2581

```

Metode Cramer

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 4
Input how many columns the matrix will have (enter a positive integer): 5
Enter your matrix with 4x5 matrix size.
8 1 3 2 0
2 9 -1 -2 1
1 3 2 -1 2
1 0 6 4 3

Here is the matrix read by the program:
8.000000    1.000000    3.000000    2.000000    0.000000
2.000000    9.000000   -1.000000   -2.000000    1.000000
1.000000    3.000000    2.000000   -1.000000    2.000000
1.000000    0.000000    6.000000    4.000000    3.000000

The result of Cramer method for the system of linear equation is the following :
X0 bernilai -0.2243
X1 bernilai 0.1824
X2 bernilai 0.7095
X3 bernilai -0.2581

```

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase3-a.txt

Here is the matrix read by the program:
8.000000    1.000000    3.000000    2.000000    0.000000
2.000000    9.000000   -1.000000   -2.000000    1.000000
1.000000    3.000000    2.000000   -1.000000    2.000000
1.000000    0.000000    6.000000    4.000000    3.000000

The result of Cramer method for the system of linear equation is the following :
X0 bernilai -0.2243
X1 bernilai 0.1824
X2 bernilai 0.7095
X3 bernilai -0.2581

```

b.

$$\begin{aligned}
 x_7 + x_8 + x_9 &= 13.00 \\
 x_4 + x_5 + x_6 &= 15.00 \\
 x_1 + x_2 + x_3 &= 8.00 \\
 0.04289(x_3 + x_5 + x_7) + 0.75(x_6 + x_8) + 0.61396x_9 &= 14.79 \\
 0.91421(x_3 + x_5 + x_7) + 0.25(x_2 + x_4 + x_6 + x_8) &= 14.31 \\
 0.04289(x_3 + x_5 + x_7) + 0.75(x_2 + x_4) + 0.61396x_1 &= 3.81 \\
 x_3 + x_6 + x_9 &= 18.00 \\
 x_2 + x_5 + x_8 &= 12.00 \\
 x_1 + x_4 + x_7 &= 6.00 \\
 0.04289(x_1 + x_5 + x_9) + 0.75(x_2 + x_6) + 0.61396x_3 &= 10.51 \\
 0.91421(x_1 + x_5 + x_9) + 0.25(x_2 + x_4 + x_6 + x_8) &= 16.13 \\
 0.04289(x_1 + x_5 + x_9) + 0.75(x_4 + x_8) + 0.61396x_7 &= 7.04
 \end{aligned}$$

Metode Gauss

input terminal

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 12
Input how many columns the matrix will have (enter a positive integer): 10
Enter your matrix with 12x10 matrix size.
0 0 0 0 0 0 1 1 1 13
0 0 0 1 1 1 0 0 0 15
1 1 1 0 0 0 0 0 0 8
0 0 0.04289 0 0.04289 0.75 0.04289 0.75 0.61396 14.79
0 0.25 0.91421 0.25 0.91421 0.25 0.91421 0.25 0 14.31
0.61396 0.75 0.04289 0.75 0.04289 0 0.04289 0 0 3.81
0 0 1 0 0 1 0 0 1 18
0 1 0 0 1 0 0 1 0 12
1 0 0 1 0 0 1 0 0 6
0.04289 0.75 0.61396 0 0.04289 0.75 0 0 0.04289 10.51
0.91421 0.25 0 0.25 0.91421 0.25 0 0.25 0.91421 16.13
0.04289 0 0 0.75 0.04289 0 0.61396 0.75 0.04289 7.04

Here is the matrix read by the program:
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.000000 1.000000 1.000000 13.000000
0.000000 0.000000 0.000000 1.000000 1.000000 0.000000 0.000000 0.000000 0.000000 15.000000
1.000000 1.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 8.000000
0.000000 0.000000 0.042890 0.000000 0.042890 0.750000 0.042890 0.750000 0.613960 14.790000
0.000000 0.250000 0.914210 0.000000 0.914210 0.250000 0.914210 0.250000 0.000000 14.310000
0.613960 0.750000 0.042890 0.750000 0.042890 0.000000 0.042890 0.000000 0.000000 3.810000
0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 1.000000 0.000000 1.000000 18.000000
0.000000 1.000000 0.000000 0.000000 0.000000 1.000000 0.000000 1.000000 0.000000 12.000000
1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 6.000000
0.042890 0.750000 0.613960 0.000000 0.042890 0.750000 0.000000 0.000000 0.042890 10.510000
0.914210 0.250000 0.000000 0.250000 0.914210 0.250000 0.000000 0.250000 0.914210 16.130000
0.042890 0.000000 0.000000 0.750000 0.042890 0.000000 0.613960 0.750000 0.042890 7.040000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000 0.000000 0.000000 17.486594 1.000000 0.000000 14.314759 17.486594 1.000000 164.140825
0.000000 1.000000 0.000000 -62.945675 0.000000 1.000000 -52.346782 -62.945675 0.000000 -535.716736
0.000000 0.000000 1.000000 75.671471 0.000000 0.000000 62.945675 75.671471 0.000000 660.071588
0.000000 0.000000 0.000000 1.000000 -0.016807 -0.022073 0.818613 0.977927 -0.018070 8.138034
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.105520 0.442296 1.025293 22.964932
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.319695 -0.762507 0.010965 5.110999
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.000000 1.000000 18.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.192619 6.965506
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 5.000770
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000

There are no solution.

```

Metode Gauss-Jordan

input file

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase3-b.txt

Here is the matrix read by the program:
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.000000 1.000000 1.000000 13.000000
0.000000 0.000000 0.000000 1.000000 1.000000 0.000000 0.000000 0.000000 0.000000 15.000000
1.000000 1.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 8.000000
0.000000 0.000000 0.042890 0.000000 0.042890 0.750000 0.042890 0.750000 0.613960 14.790000
0.000000 0.250000 0.914210 0.000000 0.914210 0.250000 0.914210 0.250000 0.000000 14.310000
0.613960 0.750000 0.042890 0.750000 0.042890 0.000000 0.042890 0.000000 0.000000 3.810000
0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 18.000000
0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.000000 12.000000
1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 6.000000
0.042890 0.750000 0.613960 0.000000 0.042890 0.750000 0.000000 0.000000 0.042890 10.510000
0.914210 0.250000 0.000000 0.250000 0.914210 0.250000 0.000000 0.250000 0.914210 16.130000
0.042890 0.000000 0.000000 0.750000 0.042890 0.000000 0.613960 0.750000 0.042890 7.040000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 1.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000 0.000000

There are no solution.

```


Metode Matriks Balikan

input terminal

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 12
Input how many columns the matrix will have (enter a positive integer): 10
Enter your matrix with 12x10 matrix size.
0 0 0 0 0 1 1 1 13
0 0 0 1 1 1 0 0 15
1 1 1 0 0 0 0 0 8
0 0 0.04289 0 0.04289 0.75 0.04289 0.75 0.61396 14.79
0 0.25 0.91421 0.25 0.91421 0.25 0.91421 0.25 0 14.31
0.61396 0.75 0.04289 0.75 0.04289 0 0.04289 0 0 3.81
0 0 1 0 0 1 0 0 1 18
0 1 0 0 1 0 0 1 0 12
1 0 0 1 0 0 1 0 0 6
0.04289 0.75 0.61396 0 0.04289 0.75 0 0 0.04289 10.51
0.91421 0.25 0 0.25 0.91421 0.25 0 0.25 0.91421 16.13
0.04289 0 0 0.75 0.04289 0 0.61396 0.75 0.04289 7.04

Here is the matrix read by the program:
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.000000 1.000000 13.000000
0.000000 0.000000 0.000000 1.000000 1.000000 0.000000 0.000000 0.000000 15.000000
1.000000 1.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 8.000000
0.000000 0.000000 0.042890 0.000000 0.042890 0.750000 0.042890 0.750000 0.613960
0.000000 0.250000 0.914210 0.250000 0.914210 0.250000 0.914210 0.250000 0.000000
0.613960 0.750000 0.042890 0.750000 0.042890 0.000000 0.042890 0.000000 0.000000
0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 1.000000 18.000000
0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 12.000000
1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 6.000000
0.042890 0.750000 0.613960 0.000000 0.042890 0.750000 0.000000 0.000000 0.042890
0.914210 0.250000 0.000000 0.250000 0.914210 0.250000 0.000000 0.250000 0.914210
0.042890 0.000000 0.000000 0.750000 0.042890 0.000000 0.613960 0.750000 0.042890

The result of Inverse Method (AX=B -> X=(A^-1)B) method for the system of linear equation is the following :
This SPL cant be finished using Inverse Matrix method, because the SPL is not on square matrix format.
```

Metode Cramer

input file

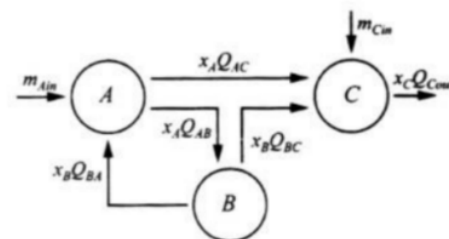
```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase3-b.txt

Here is the matrix read by the program:
0.000000 0.000000 0.000000 0.000000 0.000000 1.000000 1.000000 1.000000 13.000000
0.000000 0.000000 0.000000 1.000000 1.000000 0.000000 0.000000 0.000000 15.000000
1.000000 1.000000 1.000000 0.000000 0.000000 0.000000 0.000000 0.000000 8.000000
0.000000 0.000000 0.042890 0.000000 0.042890 0.750000 0.042890 0.750000 0.613960
0.000000 0.250000 0.914210 0.250000 0.914210 0.250000 0.914210 0.250000 0.000000
0.613960 0.750000 0.042890 0.750000 0.042890 0.000000 0.042890 0.000000 0.000000
0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 1.000000 18.000000
0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 12.000000
1.000000 0.000000 0.000000 1.000000 0.000000 0.000000 1.000000 0.000000 6.000000
0.042890 0.750000 0.613960 0.000000 0.042890 0.750000 0.000000 0.000000 0.042890
0.914210 0.250000 0.000000 0.250000 0.914210 0.250000 0.000000 0.250000 0.914210
0.042890 0.000000 0.000000 0.750000 0.042890 0.000000 0.613960 0.750000 0.042890

The result of Cramer method for the system of linear equation is the following :
This SPL cant be finished using cramer method, because the SPL is not on square matrix format.
```

4.4 Lihatlah sistem reaktor pada gambar berikut



Dengan laju volume Q dalam m^3/s dan input massa min dalam mg/s . Konservasi massa pada tiap inti reaktor adalah sebagai berikut:

$$A: m_{Ain} + Q_{BA}x_B - Q_{AB}x_A - Q_{AC}x_A = 0$$

$$B: Q_{AB}x_A - Q_{BA}x_B - Q_{BC}x_B = 0$$

$$C: m_{Cin} + Q_{AC}x_A + Q_{BC}x_B - Q_{Cout}x_C = 0$$

Tentukan solusi x_A , x_B , x_C dengan menggunakan parameter berikut : $Q_{AB} = 40$, $Q_{AC} = 80$, $Q_{BA} = 60$, $Q_{BC} = 20$ dan $Q_{Cout} = 150 m^3/s$ dan $m_{Ain} = 1300$ dan $m_{Cin} = 200 mg/s$.

Metode gauss

```

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 3
Input how many columns the matrix will have (enter a positive integer): 4
Enter your matrix with 3x4 matrix size.
-120 60 0 -1300
40 -80 0 0
80 20 -150 -200

Here is the matrix read by the program:
-120.000000    60.000000    0.000000    -1300.000000
40.000000     -80.000000    0.000000    0.000000
80.000000     20.000000   -150.000000   -200.000000

The result of Gauss method for the system of linear equation is the following :

This is the the OBE result:
1.000000    -0.500000    0.000000    10.833333
0.000000     1.000000    0.000000     7.222222
0.000000     0.000000     1.000000    10.000000

x1 = 14.4444
x2 = 7.2222
x3 = 10.0000

```

Metode gauss Jordan

```

How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase4.txt

Here is the matrix read by the program:
-120.000000    60.000000    0.000000    -1300.000000
40.000000     -80.000000    0.000000     0.000000
80.000000     20.000000   -150.000000   -200.000000

The result of Gauss-Jordan method for the system of linear equation is the following :

This is the the OBE result:
1.000000     0.000000     0.000000    14.444444
0.000000     1.000000     0.000000     7.222222
0.000000     0.000000     1.000000    10.000000

x1 = 14.4444
x2 = 7.2222
x3 = 10.0000

```

Metode Matriks balikan

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 1
Input how many rows the matrix will have (enter a positive integer): 3
Input how many columns the matrix will have (enter a positive integer): 4
Enter your matrix with 3x4 matrix size.
-120 60 0 -1300
40 -80 0 0
80 20 -150 -200

Here is the matrix read by the program:
-120.000000    60.000000    0.000000   -1300.000000
40.000000     -80.000000    0.000000    0.000000
80.000000     20.000000   -150.000000   -200.000000

The result of Inverse Method ( $AX=B \rightarrow X=(A^{-1})B$ ) method for the system of linear equation is the following :
X1 bernilai 14.4444
X2 bernilai 7.2222
X3 bernilai 10.0000
```

Metode Cramer

```
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt: testcase4.txt

Here is the matrix read by the program:
-120.000000    60.000000    0.000000   -1300.000000
40.000000     -80.000000    0.000000    0.000000
80.000000     20.000000   -150.000000   -200.000000

The result of Cramer method for the system of linear equation is the following :
X0 bernilai 14.4444
X1 bernilai 7.2222
X2 bernilai 10.0000
```

4.5 Studi Kasus Interpolasi

Gunakan tabel di bawah ini untuk mencari polinom interpolasi dari pasangan titik-titik yang terdapat dalam tabel. Program menerima masukan nilai x yang akan dicari nilai fungsi $f(x)$.

a.

x	0.1	0.3	0.5	0.7	0.9	1.1	1.3
$f(x)$	0.003	0.067	0.148	0.248	0.370	0.518	0.697

Lakukan pengujian pada nilai-nilai berikut:

$$x = 0.2 \quad f(x) = ?$$

$$x = 0.55 \quad f(x) = ?$$

$$x = 0.85 \quad f(x) = ?$$

$$x = 1.28 \quad f(x) = ?$$

$x = 0.2$

```

=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-a.txt
f(x) = -0.02297656250000014 + 0.240000000000000246x^1 + 0.19739583333333196x^2 + 3.4409860425904843E-14x^3 + 0.026041666666624073x^4 + 2.550133859867565E-14x^5 + -5.910407676398067E-15x^6
f(0.2) = -0.02297656250000014 + 0.240000000000000246(0.2)^1 + 0.19739583333333196(0.2)^2 + 3.4409860425904843E-14(0.2)^3 + 0.026041666666624073(0.2)^4 + 2.550133859867565E-14(0.2)^5 + -5.910407676398067E-15(0.2)^6
Hasil taksiran adalah : 0.032961

```

$x = 0.55$

```

=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-a.txt
f(x) = -0.02297656250000014 + 0.240000000000000246x^1 + 0.19739583333333196x^2 + 3.4409860425904843E-14x^3 + 0.026041666666624073x^4 + 2.550133859867565E-14x^5 + -5.910407676398067E-15x^6
f(0.55) = -0.02297656250000014 + 0.240000000000000246(0.55)^1 + 0.19739583333333196(0.55)^2 + 3.4409860425904843E-14(0.55)^3 + 0.026041666666624073(0.55)^4 + 2.550133859867565E-14(0.55)^5 + -5.910407676398067E-15(0.55)^6
Hasil taksiran adalah : 0.17119

```

$x = 0.85$

```

=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-a.txt
f(x) = -0.02297656250000014 + 0.240000000000000246x^1 + 0.19739583333333196x^2 + 3.4409860425904843E-14x^3 + 0.026041666666624073x^4 + 2.550133859867565E-14x^5 + -5.910407676398067E-15x^6
f(0.85) = -0.02297656250000014 + 0.240000000000000246(0.85)^1 + 0.19739583333333196(0.85)^2 + 3.4409860425904843E-14(0.85)^3 + 0.026041666666624073(0.85)^4 + 2.550133859867565E-14(0.85)^5 + -5.910407676398067E-15(0.85)^6
Hasil taksiran adalah : 0.337236

```

$x = 1.28$

```

=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-a.txt
f(x) = -0.02297656250000014 + 0.240000000000000246x^1 + 0.19739583333333196x^2 + 3.4409860425904843E-14x^3 + 0.026041666666624073x^4 + 2.550133859867565E-14x^5 + -5.910407676398067E-15x^6
f(1.28) = -0.02297656250000014 + 0.240000000000000246(1.28)^1 + 0.19739583333333196(1.28)^2 + 3.4409860425904843E-14(1.28)^3 + 0.026041666666624073(1.28)^4 + 2.550133859867565E-14(1.28)^5 + -5.910407676398067E-15(1.28)^6
Hasil taksiran adalah : 0.677542

```

b.

Jumlah kasus positif baru Covid-19 di Indonesia semakin fluktuatif dari hari ke hari. Di bawah ini diperlihatkan jumlah kasus baru Covid-19 di Indonesia mulai dari tanggal 17 Juni 2022 hingga 31 Agustus 2022:

Tanggal	Tanggal (desimal)	Jumlah Kasus Baru
17/06/2022	6,567	12.624
30/06/2022	7	21.807
08/07/2022	7,258	38.391
14/07/2022	7,451	54.517
17/07/2022	7,548	51.952
26/07/2022	7,839	28.228
05/08/2022	8,161	35.764
15/08/2022	8,484	20.813
22/08/2022	8,709	12.408
31/08/2022	9	10.534

Tanggal (desimal) = bulan + (tanggal / jumlah hari pada bulan tersebut)

Pada tanggal 16/07/2022

tanggal = 7.516

```

=====
Welcome to our first Algea Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-b-x1.txt
f(x) = 7.187066071658637E12 + -9.346993079172963E12x*1 + 5.334203055240283E12x*2 + -1.7568101863613738E12x*3 + 3.685508071755316E11x*4 + -5.1131876760132576E10x*5 + 4.695806315428787E9x*6 + -2.7547452066926E8x*7 + 9372849.23910132x*8 + -140993.71224863594x*9
f(7.516) = 7.187066071658637E12 + -9.346993079172963E12(7.516)*1 + 5.334203055240283E12(7.516)^2 + -1.7568101863613738E12(7.516)^3 + 3.685508071755316E11(7.516)^4 + -5.1131876760132576E10(7.516)^5 + 4.695806315428787E9(7.516)^6 + -2.7547452066926E8(7.516)^7 + 9372849.23910132(7.516)^8 + -140993.71224863594(7.516)^9
Hasil taksiran adalah : 53537.683594

```

Pada tanggal 10/08/2022

tanggal = 8.322

```

=====
Welcome to our first Algea Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-b-x2.txt
f(x) = 7.187066071658637E12 + -9.346993079172963E12x*1 + 5.334203055240283E12x*2 + -1.7568101863613738E12x*3 + 3.685508071755316E11x*4 + -5.1131876760132576E10x*5 + 4.695806315428787E9x*6 + -2.7547452066926E8x*7 + 9372849.23910132x*8 + -140993.71224863594x*9
f(8.322) = 7.187066071658637E12 + -9.346993079172963E12(8.322)*1 + 5.334203055240283E12(8.322)^2 + -1.7568101863613738E12(8.322)^3 + 3.685508071755316E11(8.322)^4 + -5.1131876760132576E10(8.322)^5 + 4.695806315428787E9(8.322)^6 + -2.7547452066926E8(8.322)^7 + 9372849.23910132(8.322)^8 + -140993.71224863594(8.322)^9
Hasil taksiran adalah : 36343.644531

```

Pada tanggal 05/09/2022

tanggal = 9.166

```

=====
Welcome to our first Algea Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polynomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-b-x3.txt
f(x) = 7.187066071658637E12 + -9.346993079172963E12x*1 + 5.334203055240283E12x*2 + -1.7568101863613738E12x*3 + 3.685508071755316E11x*4 + -5.1131876760132576E10x*5 + 4.695806315428787E9x*6 + -2.7547452066926E8x*7 + 9372849.23910132x*8 + -140993.71224863594x*9
f(9.166) = 7.187066071658637E12 + -9.346993079172963E12(9.166)*1 + 5.334203055240283E12(9.166)^2 + -1.7568101863613738E12(9.166)^3 + 3.685508071755316E11(9.166)^4 + -5.1131876760132576E10(9.166)^5 + 4.695806315428787E9(9.166)^6 + -2.7547452066926E8(9.166)^7 + 9372849.23910132(9.166)^8 + -140993.71224863594(9.166)^9
Hasil taksiran adalah : -659015.265625

```

pada tanggal = 3.843

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polinomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-b.xd.txt
f(x) = 7.187066071658637E12 + -9.346993079172963E12x^1 + 5.334203055240283E12x^2 + -1.7568101863613738E12x^3 + 3.685508071755316E11x^4 + -5.1131876760132576E10x^5 + 4.695806315428787E9x^6 + -2.7547453
942066926E8x^7 + 9372849.23910132x^8 + -140993.71224863594x^9
f(3.483) = 7.187066071658637E12 + -9.346993079172963E12(3.483)^1 + 5.334203055240283E12(3.483)^2 + -1.7568101863613738E12(3.483)^3 + 3.685508071755316E11(3.483)^4 + -5.1131876760132576E10(3.483)^5 + 4
.695806315428787E9(3.483)^6 + -2.7547453942066926E8(3.483)^7 + 9372849.23910132(3.483)^8 + -140993.71224863594(3.483)^9
Hasil taksiran adalah : 3663469525.131348
```

Sederhanakan fungsi $f(x)$ yang memenuhi kondisi

c.
$$f(x) = \frac{x^2 + \sqrt{x}}{e^x + x}$$

dengan polinom interpolasi derajat n di dalam selang $[0, 2]$.

Sebagai contoh, jika $n = 5$, maka titik-titik x yang diambil di dalam selang $[0, 2]$ berjarak $h = (2 - 0)/5 = 0.4$.

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to polinomial interpolation !
=====
How do you want to input your coordinates?
1. By inputting manually via program.
2. By reading .txt file

Which option do you like to proceed? 2
Input the filename and don't forget to include .txt : testcase5-c.txt
f(x) = 2.03525675000000065x^1 + -3.55267916666666973x^2 + 3.2371100260417136x^3 + -1.4212630208333623x^4 + 0.2362556966145896x^5
f(0.5) = 2.03525675000000065(0.5)^1 + -3.55267916666666973(0.5)^2 + 3.2371100260417136(0.5)^3 + -1.4212630208333623(0.5)^4 + 0.2362556966145896(0.5)^5
Hasil taksiran adalah : 0.452651
```

4.6 Studi Kasus Linear Berganda

Table 12.1: Data for Example 12.1

Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3	Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3
0.90	72.4	76.3	29.18	1.07	23.2	76.8	29.38
0.91	41.6	70.3	29.35	0.94	47.4	86.6	29.35
0.96	34.3	77.1	29.24	1.10	31.5	76.9	29.63
0.89	35.1	68.0	29.27	1.10	10.6	86.3	29.56
1.00	10.7	79.0	29.78	1.10	11.2	86.0	29.48
1.10	12.9	67.4	29.39	0.91	73.3	76.3	29.40
1.15	8.3	66.8	29.69	0.87	75.4	77.9	29.28
1.03	20.1	76.9	29.48	0.78	96.6	78.7	29.29
0.77	72.2	77.7	29.09	0.82	107.4	86.8	29.03
1.07	24.0	67.7	29.60	0.95	54.9	70.9	29.37

Source: Charles T. Hare, "Light-Duty Diesel Emission Correction Factors for Ambient Conditions," EPA-600/2-77-116. U.S. Environmental Protection Agency.

Gunakan *Normal Estimation Equation for Multiple Linear Regression* untuk mendapatkan regresi linear berganda dari data pada tabel di atas, kemudian estimasi nilai Nitrous Oxide apabila Humidity bernilai 50%, temperatur 76°F, dan tekanan udara sebesar 29.30.

Memprediksi $x_1 = 50$, $x_2 = 76$, $x_3 = 29.30$

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to Multiple Linear Regression !
=====
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Masukkan Pilihan: 2
Input the filename and don't forget to include .txt : eqweq
Please include '.txt' in the filename. Re-enter the filename: .txt
File not found in test folder. Make sure your file is existed in that folder.
Input the filename and don't forget to include .txt: testcase6.txt

Normal Estimation Equation for Multiple Linear Regression :
20.0000b0 + 863.1000b1 + 1530.4000b2 + 587.8400b3 = 19.4200
863.1000b0 + 54876.8900b1 + 67000.0900b2 + 25283.3950b3 = 779.4770
1530.4000b0 + 67000.0900b1 + 117912.3200b2 + 44976.8670b3 = 1483.4370
587.8400b0 + 25283.3950b1 + 44976.8670b2 + 17278.5086b3 = 571.1219

This is the augmented matrix of this linear regression

20.000000      863.100000      1530.400000      587.840000      19.420000
863.100000      54876.890000      67000.090000      25283.395000      779.477000
1530.400000      67000.090000      117912.320000      44976.867000      1483.437000
587.840000      25283.395000      44976.867000      17278.508600      571.121900

This is the result of gauss method for the augmented matrix above :
1.000000      43.155000      76.520000      29.392000      0.971000
0.000000      1.000000      0.054208      -0.004812      -0.003324
0.000000      0.000000      1.000000      -0.000067      0.000789
0.000000      0.000000      0.000000      1.000000      0.154155

The linear regression equation : y = -3.5078 + -0.0026x1 + 0.0008x2 + 0.1542x3
Please input variable value that want to be predicted:
x1 = 50
x2 = 76
x3 = 29.3
The result of estimating the function value of the x values is
f(50.0000,76.0000,29.3000) = 0.9384
```

4.7 Studi Kasus Bicubic Linear Interpolation

Diberikan matriks input dengan bentuk sebagai berikut. Format matriks masukan bukan mewakili nilai matriks, tetapi mengikuti format masukan pada bagian “Spesifikasi Tugas” nomor 7.

$$\begin{pmatrix} 21 & 98 & 125 & 153 \\ 51 & 101 & 161 & 59 \\ 0 & 42 & 72 & 210 \\ 16 & 12 & 81 & 96 \end{pmatrix}$$

Tentukan nilai:

$$\begin{aligned} f(0, 0) &= ? \\ f(0.5, 0.5) &= ? \\ f(0.25, 0.75) &= ? \\ f(0.1, 0.9) &= ? \end{aligned}$$

f(0,0)

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to Bicubic Linear Interpolation !
=====
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Masukan Pilihan: 2
Input the filename that contains derived matrix, a and b values.Don't forget to include .txt: testcase7-x1.txt

Matrix Inputan:
21.000000      98.000000      125.000000      153.000000
51.000000      101.000000     161.000000      59.000000
0.000000       42.000000      72.000000     210.000000
16.000000      12.000000      81.000000      96.000000

nilai a dan b adalah:
0.0,0.0

Matriks Turunan setelah di ubah bentuknya:
21.000000
98.000000
125.000000
153.000000
51.000000
101.000000
161.000000
59.000000
0.000000
42.000000
72.000000
210.000000
16.000000
12.000000
81.000000
96.000000

Nilai a hasil baca file adalah: 0.0
Nilai b hasil baca file adalah: 0.0

Hasil Intepolasi Spline Bicubic dari f(0.0,0.0) adalah 21.0
```


$f(0.5,0.5)$

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to Bicubic Linear Interpolation !
=====
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Masukan Pilihan: 1
Please input 4 x 4 derived matrix :
21 98 125 153
51 101 161 59
0 42 72 210
16 12 81 96

Derived matrix after trasform shape:
21.000000
98.000000
125.000000
153.000000
51.000000
101.000000
161.000000
59.000000
0.000000
42.000000
72.000000
210.000000
16.000000
12.000000
81.000000
96.000000

Please enter a value : 0.5

Please enter b value : 0.5

Hasil Intepolasi Spline Bicubic dari  $f(0.5,0.5)$  adalah 87.796875
```

$f(0.25,0.75)$

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to Bicubic Linear Interpolation !
=====
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Masukan Pilihan: 2
Input the filename that contains derived matrix, a and b values.Don't forget to include .txt: testcase7-x3.txt

Matrix Inputan:
21.000000    98.000000    125.000000    153.000000
51.000000    101.000000    161.000000    59.000000
0.000000     42.000000     72.000000    210.000000
16.000000    12.000000     81.000000     96.000000

nilai a dan b adalah:
0.25,0.75

Matriks Turunan setelah di ubah bentuknya:
21.000000
98.000000
125.000000
153.000000
51.000000
101.000000
161.000000
59.000000
0.000000
42.000000
72.000000
210.000000
16.000000
12.000000
81.000000
96.000000

Nilai a hasil baca file adalah: 0.25

Nilai b hasil baca file adalah: 0.75

Hasil Intepolasi Spline Bicubic dari  $f(0.25,0.75)$  adalah 117.732177734375
```

f(0.1,0.9)

```
=====
Welcome to our first Algeo Project
This is JUN HOK 88 's hardwork.
=====
Welcome to Bicubic Linear Interpolation !
=====
How do you want to input your matrix?
1. By inputting manually via program.
2. By reading .txt file

Masukan Pilihan: 1
Please input 4 x 4 derived matrix :
21 98 125 153
51 101 161 59
0 42 72 210
16 12 81 96

Derived matrix after trasform shape:
21.000000
98.000000
125.000000
153.000000
51.000000
101.000000
161.000000
59.000000
0.000000
42.000000
72.000000
210.000000
16.000000
12.000000
81.000000
96.000000

Please enter a value : 0.1

Please enter b value : 0.9

Hasil Intepolasi Spline Bicubic dari f(0.1,0.9) adalah 128.57518700000003
```

BAB 5

PENUTUP

1. Kesimpulan

Pada tugas besar kali ini, kami berusaha semaksimal mungkin untuk mengimplementasikan seluruh materi mata kuliah Aljabar Linier dan Geometri IF2123 yang kami pelajari di kelas. Kami melakukan implementasi materi-materi tersebut dan dituangkan ke dalam bentuk program dalam bahasa Java. Program kami dapat menyelesaikan persoalan-persoalan yang menggunakan matriks, seperti menyelesaikan Sistem Persamaan Linear (SPL), menghitung nilai determinan dari sebuah matriks, menentukan matriks balikan dari sebuah matriks, menghitung interpolasi polinomial, regresi linear berganda, dan interpolasi *bicubic spline*.

Dengan program penyelesaian Sistem Persamaan Linear (SPL) ini, pengguna dapat dibantu dalam menyelesaikan permasalahan SPL, bahkan ketika jumlah variabelnya banyak. Program ini dapat menyelesaikan interpolasi yang akan membantu pengguna dalam menaksir nilai-nilai dengan data yang terbatas menggunakan metode interpolasi polinomial, regresi linear berganda, dan interpolasi *bicubic spline*.

Melalui tugas besar ini, kami membuat program yang membantu penggunanya untuk menyelesaikan permasalahan matematika atau permasalahan riilnya untuk diselesaikan dengan cepat dan akurat sehingga permasalahan-permasalahan yang sangat sulit untuk diselesaikan secara manual, dapat diselesaikan dengan cepat oleh program yang telah kami buat.

2. Saran

- Kode seharusnya dibuat lebih bersih dan sesuai perjanjian antar anggota kelompok agar tidak kebingungan ketika melakukan integrasi fungsi yang dibuat satu anggota dengan anggota lainnya.

- Tidak melakukan pengecekan *test case* secara masif hanya pada saat dekat deadline pengumpulan tugas besar karena terdapat banyak *hidden-bug* yang baru muncul ketika sudah di akhir masa pengumpulan.
- Nama fungsi tidak konsisten, ada yang menggunakan bahasa Inggris dan ada yang menggunakan bahasa Indonesia karena kebiasaan yang berbeda antar anggota.
- *Timeline* pengerjaan tugas besar harusnya telah ditentukan sesaat perilis tugas besar ini agar tidak kewalahan ketika sudah mendekati deadline pengumpulan tugas besar.
- Koordinasi harus lebih jelas ketika menggunakan *respository* github karena ketidakdisiplinan para anggota dalam menggunakannya menyebabkan sering terjadinya *conflict* yang menghambat pekerjaan kelompok.
- Struktur folder seharusnya telah dibuat dengan jelas sesaat dibuatnya github dibandingkan membuatnya ketika masa *development* karena itu mungkin juga menyebabkan bug mayor pada program.

3. Refleksi

Melalui tugas besar ini, kami mendapatkan berbagai ilmu yang beragam, pengalaman yang unik, dan sensasi-sensasi baru yang belum pernah dirasakan. Kami juga melatih kemampuan berpikir komputasional untuk melakukan pembuatan struktur program dari awal, kemampuan bekerjasama dan berkomunikasi sebagai satu kelompok, kemampuan untuk menggunakan git untuk melakukan *update* pada pekerjaan kami, kemampuan melakukan *debugging* pada error yang belum pernah dialami sebelumnya, kemampuan untuk meng-*handle* error yang kemungkinan terjadi pada program. Dengan tugas besar ini juga, kami menjadi sadar bahwa bahasa pemrograman yang baru bukanlah sebuah tantangan yang besar dibandingkan program utama yang hendak dikembangkan.

4. Komentar

Dari Elbert Chailes : Makasihh semua yang sudah berkontribusi.

Dari Benardo : Gass tubess ke dua.

Dari Kharris Khissunica : Mantap.

5. Link Repository

Link repository kelompok kami untuk tugas besar pertama mata kuliah IF 2123 Aljabar Linier dan Geometri adalah sebagai berikut.

Link : <https://github.com/ChaiGans/Algeo01-22045>

REFERENSI

<https://www.w3schools.com/java/>

<https://www.geeksforgeeks.org/different-ways-reading-text-file-java/>

<https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/algeo23-24.htm>

https://www.w3schools.com/java/java_try_catch.asp

https://en.wikipedia.org/wiki/Bicubic_interpolation

https://en.wikipedia.org/wiki/Polynomial_interpolation

<https://www.geeksforgeeks.org/throw-throws-java/>