LAPORAN TUGAS BESAR 1
IF2123 ALJABAR LINEAR DAN GEOMETRI
SISTEM PERSAMAAN LINEAR, DETERMINAN DAN APLIKASINYA

Kelompok 22 - SPL deck

Anggota Kelompok:

Fajar Maulana Herawan	**13521080**
Vieri Fajar Firdaus	**13521099**
Mohammad Farhan Fahrezy	**13521106**

PROGRAM STUDI TEKNIK INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

2022

Daftar Isi

BAB I
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 -1b), 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.

BAB II
Teori Singkat

2.1. Metode Eliminasi Gauss

Eliminasi Gauss diperkenalkan Karl Friendrich Gauss (1777 – 1855) dengan meng-augmentasi matriks A dengan matriks B, lalu matriks gabungan tersebut diubah menjadi suatu matriks eselon baris dengan memanfaatkan Operasi Baris Elementer (OBE). Operasi yang diliput oleh OBE adalah penjumlahan atau pengurangan suatu baris dengan k kali baris yang lain (dengan k bebas) dan operasi tukar baris.

Suatu matriks dikatakan memiliki bentuk eselon baris jika :

1) Entri bukan nol pertama dalam setiap baris adalah 1.
2) Jika baris tidak seluruhnya mengandung nol, maka banyaknya entri nol di bagian muka pada baris lebih besar dari banyaknya entri nol di bagian muka pada baris.
3) Jika terdapat baris-baris yang entrinya semuanya nol, maka baris - baris ini berada dibawah baris-baris yang memiliki entri-entri nol.

$$A = \begin{bmatrix} 0 & 1 & 3 & 7 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Contoh matriks eselon baris

Setelah matriks berhasil diubah menjadi matriks Eselon Baris, maka dapat digunakan substitusi untuk mendapatkan solusi tiap variabel.

2.2. Metode Eliminasi Gauss-Jordan

Metode Eliminasi Gauss-Jordan dicetuskan oleh Wilhelm Jordan, seorang insinyur Jerman, pada tahun 1887 dengan cara mengembangkan metode Eliminasi Gauss. Langkah - langkah metode ini sama seperti metode Eliminasi Gauss, namun setelah matriks berhasil diubah menjadi matriks eselon baris, perlu dilanjutkan untuk mengubah matriks tersebut menjadi matriks eselon baris tereduksi.

Suatu matriks dikatakan memiliki bentuk eselon baris tereduksi jika :

1) Memenuhi syarat yang sama dengan eselon baris
2) Setiap kolom yang memiliki 1 utama memiliki nol di tempat lain.

$$\begin{bmatrix} 1 & 0 & 0 & 4 \\ 0 & 1 & 0 & 7 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Contoh matriks eselon baris tereduksi

Manfaat daripada menggunakan metode ini dibanding metode eliminasi Gauss biasa adalah bisa didapatkan nilai dari variabel tanpa harus melakukan proses substitusi

2.3. Determinan

Determinan adalah nilai yang dapat dihitung dari unsur suatu matriks persegi, yaitu matriks yang memiliki ukuran baris dan kolom yang sama. Determinan matriks A ditulis dengan tanda det(A), det A, atau |A|. Determinan dapat dianggap sebagai faktor penskalaan transformasi yang digambarkan oleh matriks. Metode untuk mencari nilai determinan ada 2, yaitu metode Reduksi Baris (Gauss) dan metode Ekspansi Kofaktor.

Metode Reduksi Baris dilakukan dengan melakukan eliminasi Gauss dan mencari nilai Determinan Segitiga Bawah. Hasil determinan tersebut kemudian dibagi dengan suatu peubah matriks. Nilai peubah matriks dikali -1 tiap mengganti 2 baris dan dikali k tiap mengalikan suatu baris dengan k saat melakukan Operasi Baris Elementer (OBE).

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix}, det(A) = a_{11}a_{22}a_{33}a_{44}$$

2.3.1 Determinan Segitiga Bawah

Metode Ekspansi Kofaktor dilakukan dengan mencari suatu baris yang paling banyak jumlah 0-nya dan mengalikan elemen baris tersebut dengan determinan Matriks Kofaktornya. Matriks Kofaktor suatu elemen a_{ij} adalah M_{ij} yang terdiri dari elemen Matriks selain baris i dan kolom j. Setiap Matriks Kofaktor akan dicari determinannya secara rekurens hingga mencapai kondisi basis, yaitu Matriks ukuran 2x2 dengan nilai determinannya adalah (a_{11}*a_{22}) - (a_{12}*a_{21}).

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 1 & 0 & 2 \\ 0 & 4 & 2 & 3 \\ 2 & 3 & 4 & 3 \end{bmatrix}, det(A) = -0\begin{bmatrix} 2 & 3 & 4 \\ 4 & 2 & 3 \\ 3 & 4 & 3 \end{bmatrix} + 1\begin{bmatrix} 1 & 3 & 4 \\ 0 & 2 & 3 \\ 3 & 4 & 3 \end{bmatrix} - 0\begin{bmatrix} 1 & 2 & 4 \\ 0 & 4 & 3 \\ 2 & 3 & 3 \end{bmatrix} + 2\begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 2 \\ 2 & 3 & 4 \end{bmatrix}$$

2.3.2 Metode Ekspansi Kofaktor

$$\begin{bmatrix} + & - & + & - & .. \\ - & + & - & + & .. \\ + & - & + & - & .. \\ - & + & - & + & .. \\ \vdots & \vdots & \vdots & \vdots & \end{bmatrix}$$

2.3.3 Tanda Positif dan Negatif Ekspansi Kofaktor

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, det(A) = a_{11}a_{22} - a_{12}a_{21}$$

2.3.4 Determinan Kondisi Basis (2x2)

2.4. Matriks Balikan

Suatu matriks A^{-1} merupakan sebuah balikan dari matriks A yang menghasilkan matriks identitas jika dikalikan ($A^{-1}A=I$). Matriks balikan dapat dimanfaatkan untuk mencari solusi dari Sistem Persamaan Linear yang memiliki solusi tunggal.

$$Ax = b$$

$$A^{-1}Ax = A^{-1}b$$

$$Ix = A^{-1}b$$

$$x = A^{-1}b$$

2.4.1 Solusi SPL menggunakan Matriks Balikan

Ada 2 cara untuk mendapatkan nilai balikan dari suatu matriks, yaitu metode Matriks Kofaktor dan metode Eliminasi Gauss-Jordan. Metode Matriks Kofaktor memanfaatkan adjoin dari suatu matriks dan membaginya dengan determinan matriks tersebut. Matriks adjoin merupakan transpose dari matriks kofaktor.

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

$$A^{-1} = \frac{1}{\det(A)} kof(A)^T$$

2.4.2 Metode Balikan menggunakan Matriks Kofaktor

Metode Eliminasi Gauss-Jordan memanfaatkan matriks identitas dengan membuat matriks augmented dari suatu matriks A dengan ditempelkan matriks identitas yang ukurannya sama dengan matriks A. Matriks augmented tersebut lalu diberi Eliminasi Gauss-Jordan sehingga matriks identitas berpindah ke kiri dan matriks di sebelah kanan menjadi balikan matriks A.

$$[A|I] \sim\text{(Gauss-Jordan)}\sim [I|A^{-1}]$$

2.4.3 Metode Balikan menggunakan Eliminasi Gauss-Jordan

2.5. Matriks Kofaktor

Kofaktor merupakan hasil perkalian minor dengan suatu angka yang besarnya menuruti suatu aturan yaitu $(-1)^{i+j}$, i adalah baris dan j adalah kolom. Kofaktor dari sebuah elemen baris ke-i dan kolom ke-j dari matriks A akan bisa dikenali melalui lambangnya, yaitu C_{ij}. Suatu matriks kofaktor merupakan matriks yang terbentuk dari kofaktor tiap elemen matriks tersebut.

$$Kof(A) = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \\ C_{41} & C_{42} & C_{43} & C_{44} \end{bmatrix}$$

2.5.1 Matriks Kofaktor dari Matriks berukuran 4x4

2.6. Matriks Adjoin

Matriks adjoin merupakan transpose dari suatu matriks kofaktor yang elemen-elemennya merupakan kofaktor dari elemen-elemen matriks tersebut. Adjoin dari suatu matriks A dituliskan sebagai adj(A). Matriks adjoin dapat digunakan untuk menentukan balikan dari suatu matriks A.

2.7. Kaidah Cramer

Kaidah Cramer adalah aturan yang dapat digunakan untuk menyelesaikan Sistem Persamaan Linear dengan banyak persamaan sama dengan banyak variabel, dan berlaku ketika sistem tersebut memiliki solusi yang tunggal. Jika suatu Ax=b adalah suatu SPL yang terdiri dari n persamaan linear dan n variabel sedemikian sehingga det(A)≠0, maka SPL tersebut memiliki solusi unik yaitu

$$X_1 = \frac{\det(A_1)}{\det(A)}, X_2 = \frac{\det(A_2)}{\det(A)}, \ldots, X_n = \frac{\det(A_n)}{\det(A)}$$

2.7.1 Solusi SPL dengan Kaidah Cramer

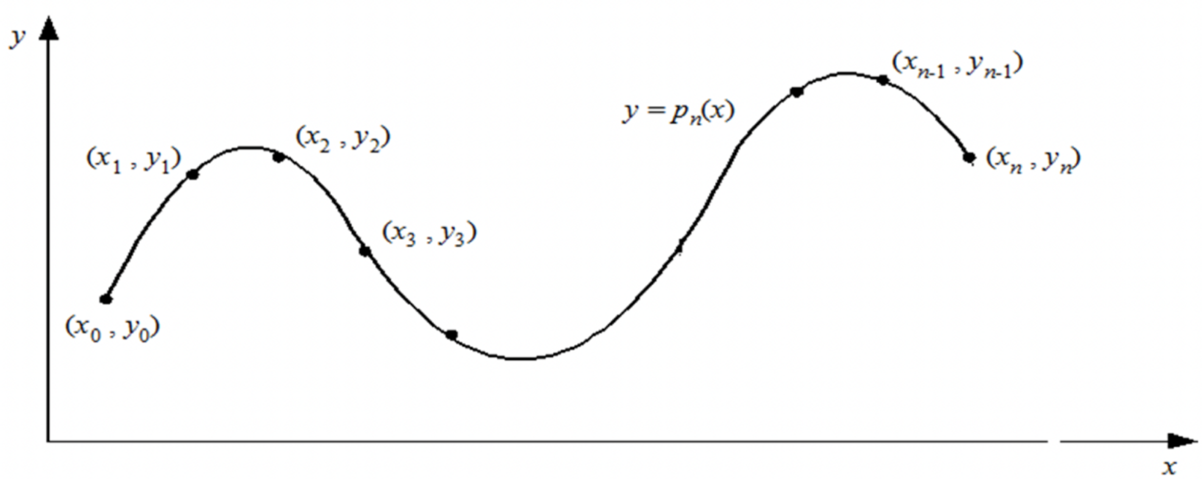
Dengan matriks A_j merupakan matriks yang diperoleh dengan mengganti entri pada kolom j dengan matriks b.

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

2.7.2 Matriks b

2.8. Interpolasi Polinom

Interpolasi Polinom adalah kaidah yang mengasumsikan pola data kita sesuai dengan polinom berderajat tinggi.



2.8.1 Grafik Persamaan Polinom

Oleh karena itu, terlebih dahulu kita tentukan persamaan polinomnya. kita dapat membuat polinom interpolasi berderajat *n* untuk *n* yang lebih tinggi asalkan tersedia (*n*+1) buah titik data. Dengan menyulihkan (x_i, y_i) ke dalam persamaan polinom

$P_n(x) = a_0 + a_1x + a_2x_2 + ... + a_nx_n$ untuk $i = 0, 1, 2, ..., n$, akan diperoleh *n* buah sistem persamaan lanjar dalam $a_0, a_1, a_2, ..., a_n$.

$$a_0 + a_1x_0 + a_2x_{02} + ... + a_nx_{on} = y_0$$

$$a_0 + a_1x_1 + a_2x_{12} + ... + a_nx_{1n} = y_1$$

$$... \dots$$

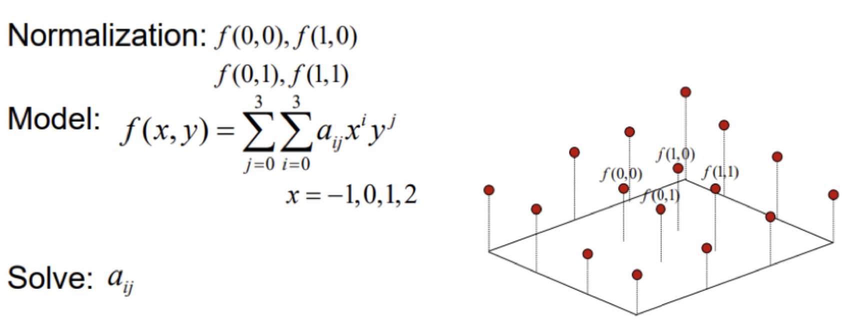
$$a_0 + a_1x_n + a_2x_{n2} + ... + a_nx_{nn} = y_n$$

2.8.2 Persamaan Polinom

Dengan menyelesaikan persamaan di atas didapatkan nilai $a_0, a_1, ..., a_n$ sehingga kita mendapatkan persamaan polinom orde *n* dan dapat menaksir nilai *x* sesuai dengan persamaan tersebut.

2.9. Interpolasi Bicubic

Interpolasi Bicubic adalah teknik interpolasi pada data 2D yang biasa digunakan dalam *image scaling*. Diberikan sebuah matrix awal, misal M, kita akan mencari persamaan interpolasi $f(x, y)$ dengan memodelkannya sebagai berikut:



2.9.1 Model Interpolasi Bicubic

Dengan melakukan substitusi nilai-nilai pada matriks 4 x 4 tersebut ke persamaaan $f(x, y)$
Akan menghasilkan sebuah matriks persamaan:

$$y = Xa$$

$$
\begin{bmatrix} f(-1,-1) \\ f(0,-1) \\ f(1,-1) \\ f(2,-1) \\ f(-1,0) \\ f(0,0) \\ f(1,0) \\ f(2,0) \\ f(-1,1) \\ f(0,1) \\ f(1,1) \\ f(2,1) \\ f(-1,2) \\ f(0,2) \\ f(1,2) \\ f(2,2) \end{bmatrix}
=
\begin{bmatrix}
1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
1 & 2 & 4 & 8 & -1 & -2 & -4 & -8 & 1 & 2 & 4 & 8 & -1 & -2 & -4 & -8 \\
1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 2 & 4 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 \\
1 & -1 & 1 & -1 & 2 & -2 & 2 & -2 & 4 & -4 & 4 & -4 & 8 & -8 & 8 & -8 \\
1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 8 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 4 & 4 & 4 & 4 & 8 & 8 & 8 & 8 \\
1 & 2 & 4 & 8 & 2 & 4 & 8 & 16 & 4 & 8 & 16 & 32 & 8 & 16 & 32 & 64
\end{bmatrix}
\begin{bmatrix} a_{00} \\ a_{10} \\ a_{20} \\ a_{30} \\ a_{01} \\ a_{11} \\ a_{21} \\ a_{31} \\ a_{02} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{03} \\ a_{13} \\ a_{23} \\ a_{33} \end{bmatrix}
$$

Elemen pada matrix X adalah koefisien a_{ij} yang diperoleh dari persamaan $f(x, y)$ di atas.Vektor a dapat dicari dari persamaan tersebut (menggunakan inverse) sehingga terbentuk fungsi interpolasi bicubic sesuai model.

2.10. Regresi Linier Berganda

Regresi Linear (akan dipelajari lebih lanjut di Probabilitas dan Statistika) merupakan salah satu metode untuk memprediksi nilai selain menggunakan Interpolasi Polinom. Meskipun sudah ada rumus jadi untuk menghitung regresi linear sederhana, terdapat rumus umum dari regresi linear yang bisa digunakan untuk regresi linear berganda, yaitu.

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

Untuk mendapatkan nilai dari setiap βi dapat digunakan Normal Estimation Equation for Multiple Linear Regression sebagai berikut:

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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

BAB III
Implementasi dalam Java

3.1. Class Matrix

Atribut

Nama	Tipe	Deskripsi
array	public double [][]	Elemen dari matriks berukuran 1000x1000
row	public int	Nilai baris efektif
col	public int	Nilai kolom efektif
peubahDeterminan	public double	Peubah determinan saat melakukan OBE

Method

Nama	Tipe	Parameter	Deskripsi
Matrix	public	-	Konstruktor untuk class Matrix dan mengisi this.array dengan elemen 0.0
IsiMatriks	public void	int **baris** int **kolom**	Mengisi matriks dengan ukuran baris x kolom
CheckZero	public int	int **index**	Mengecek banyak nol pada setiap baris
isSquare	public boolean	-	Mengecek matriks persegi
isSegitigaAtas	public boolean	-	Mengecek matriks segitiga atas
isSegitigaBawah	public boolean	-	Mengecek matriks segitiga bawah
isEmpty	public boolean	-	Mengecek matriks kosong
SwapBaris	public void	int **x** int **y**	Mengubah baris matriks indeks ke-x dengan indeks ke-y
Eselon	public boolean	-	Mengecek matriks eselon

EselonR	public boolean	-	Mengecek matriks eselon reduksi
Display	public void	-	Menampilkan matriks
CheckSolution	public boolean	-	Mengecek matriks terdapat solusi atau tidak
KaliMatriks	public void	double **pengali** int **index**	Menggali setiap elemen matriks pada baris ke-index dengan pengali
KurangmMatriks	public void	int **idx1** int **idx2** double **pengali**	Mengurangi setiap elemen matriks pada baris ke-idx1 dengan pengali dikali dengan elemen matriks pada baris ke-idx2
Hasil_OBE	public void	-	Mengubah matriks sehingga terbentuk matriks dengan hasil pengoperasian OBE
sum1	public float	int **col1**	Menjumlahkan semua elemen matriks pada kolom ke-col1
sum2	public float	int **col1** int **col2**	Menjumlahkan semua elemen matriks pada kolom ke-col1 dikali dengan kolom ke-col2
copyMatrix	public static void	Matrix **Min** Matrix **Mout**	Menyalin matriks Min ke matriks Mout
mintoZero	public void	-	Mengubah nilai -0.00 menjadi 0.00
KaliMatrix	public static Matrix	Matrix **m1** Matrix **m2**	Mengalikan matriks m1 dengan m2
transpose	public static void	Matrix **Min**	Mentrasposkan matriks Min
SolusiSPL	public static String[]	Matrix **solver**	Membuat array of string yang berisikan solusi dari matriks solver
DisplaySolution	public void	-	Menampilkan solusi matriks

3.2. Class Determinan

Atribut

- Class ini tidak memiliki atribut

Method

Nama	Tipe	Parameter	Deskripsi
DET_Reduksi_Baris_Kofaktor	public static double	Matrix **DetMat**	Menerima matriks persegi dan memberikan determinan dengan metode Ekspansi Kofaktor
DET_Gauss	public static double	Matrix **DetMat**	Menerima matriks persegi dan memberikan determinan dengan metode Eliminasi Gauss
mainKofaktor	public static void	-	Menjalankan fungsi *DET_Reduksi_Baris_Kofaktor* untuk main
mainGauss	public static void	-	Menjalankan fungsi *DET_Gauss* untuk main

3.3. Class imageScaling

Atribut

Nama	Tipe	Deskripsi
image	BufferedImage	Input image
output	BufferedImage	Output image
width	int	Lebar output image
height	int	Panjang output image
newColor	Color	Menyimpan warna RGB
c,c1,c2,c3,c4	Color	Menyimpan warna RGB
newColor,newColor1,..,newColor5	Color	Menyimpan hasil interpolasi bicubic RGB
red,green,blue,alpha,red1,greena,blue1,alpha1	int	Menyimpan hasil interpolasi bicubic RGB
resr,resg,resb,resa	Matrix	Menyimpan array input untuk diinterpolasi bicubic
nr1,ng1,nb1,...,nr4,ng4,nb4	int	Untuk mendapatkan warna

		dari gambar
hasilred	int	Menyimpan hasil interpolasi bicubic
hasilgreen	int	Menyimpan hasil interpolasi bicubic
hasilblue	int	Menyimpan hasil interpolasi bicubic

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
InputImage	public void	String **filename**	Memasukkan file yang akan diperbesar
idxundef	public void	int **i**, int **j** int **ii**, int **jj**	Merubah hasil interpolasi bicubic yang melebihi batas
idxundefmid	public void	int **i** int **j**	Merubah hasil interpolasi bicubic yang melebihi batas
MakeRGB	public void	int **i** int **j**	Merubah 1 pixel menjadi warna RGB dalam bentuk integer
makeArrayRGB	public void	int **i** int **j**	Memasukkan warna warna dari 1 pixel ke array 4x4
isigambar	public void	int **i0**, int **it** int **j0**, int **jt** int **ket**	Menginterpolasi bicubic gambar pada baris ke i0 sampai it dan kolom j0 ke jt
perbesarGambar	Public void	-	Melakukan perbesaran gambar
OutputImage	Public void	-	Memasukkan nama output file
main	Public static void	-	Menjalankan program class imageScalling

3.4. Class interpolationBicubic

Atribut

Nama	**Tipe**	**Deskripsi**
matriksInput	Public Matrix	Matriks input dari user baik keyboard maupun file.

matriks	Public Matrix	Matriks X
a	Public Double	Nilai *x* yang ingin ditaksir
b	Public Double	Nilai y yang ingin ditaksir
persamaan	Public String	Persamaan interpolasi

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
inputMatrix	Public void	-	Menerima masukan user (keyboard) dan membentuk matriksinput
hasilBicubicInterpolasi	Public void	-	Menghasilkan interpolasi bicubic sesuai model.
inputMatrixFile	Public void	String **filename**	Menerima masukan user (file) dan membentuk matriksinput
outputMatrixFile	Public Void	String **filename**	Membuat file hasil interpolasi
main	Public static void	-	Menjalankan program class interpolationBicubic

3.5. Class interpolationPolinom

Atribut

Nama	**Tipe**	**Deskripsi**
point	Public Matrix	Matrix Augmented
N	Public int	Masukkan jumlah N
nilaiTaksir	Public double	Nilai yang ingin ditaksir
konstanta	Public array of double	Nilai konstanta $a_0, a_1, a_2, ..., a_n$
persamaan	Public String	Persamaan Interpolasi

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
interpolasiInput	Public void	-	Menerima masukan user (keyboard) menghasilkan matriks augmented

persamaanPolinom	Public void	-	Menghasilkan persamaan interpolasi
nilaiTaksir	Public String	-	Menghasilkan nilai taksir dari nilai yang ingin ditaksir
interpolasiInputFile	Public void	String **filename**	Menerima masukan user (file) menghasilkan matriks augmented
interpolasiOutputFile	Public void	String **filename**	Membuat file hasil taksir dan persamaan interpolasi.
main	Public static void	-	Menjalankan program class interpolasi polinom

3.6. Class RLB

Atribut

Nama	**Tipe**	**Deskripsi**
solver	Matrix	Menyimpan array masukan
input	Scanner	Untuk input variabel
nilaiTaksir	Double	Hasil taksiran
length	int	Panjang nilai x yang menjadi taksiran
persamaan	String	Hasil persamaan dalam string

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
RLB_Ganda	Public void	-	Melakukan operasi Regresi Linear Berganda
save	Public void	-	Untuk menyimpan hasil regresi
hasiltaksir	Public double	Double **nilai**	Melakukan taksiran terhadap regresi linear berganda
inputRLB	Public void	-	Menginput nilai yang akan dilakukan regresi linear berganda
inputFile	Public void	String **filename**	Melakukan input berdasarkan file

outputFile	Public void	String **filename**	Melakukan save terhadap hasil Regresi Linear berganda dalam bentuk txt
main	public static void	-	Menjalankan program main RLB

3.7. Class SPL_Balikan

Atribut

- Class ini tidak memiliki atribut

Method

Nama	Tipe	Parameter	Deskripsi
INV_GaussJordan	public static Matrix	Matrix **inverse**	Menerima matriks persegi dan memberikan balikan dengan metode eliminasi Gauss-Jordan dengan bantuan matriks augmented identitas
Balikan_SPL	public static Matrix	Matrix **inverse**	Menerima matriks persegi dan memberikan balikan dengan metode adjoin
Hasil_INV	Public static String[]	Matrix **Axb**	Menerima matriks SPL berukuran NxN+1 dan mengeluarkan output solusi SPL dalam array of string
mainInvGauss	public static void	-	Menjalankan fungsi *INV_GaussJordan* untuk main
mainInvAdj	public static void	-	Menjalankan fungsi *Balikan_SPL* untuk main

3.8. Class SPL_Cramer

Atribut

- Class ini tidak memiliki atribut

Method

Nama	Tipe	Parameter	Deskripsi
Cramer	public static void	Matrix **MatCram**	Menerima matriks ukuran NxN+1 dan memberikan solusi SPL dengan Kaidah Cramer
main	public static	-	Menjalankan fungsi *Cramer*

	void		untuk main

3.9. Class SPL_Gauss

Atribut

- Class ini tidak memiliki atribut

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
Gauss	public static void	Matrix **solver**	Menampilkan langkah langkah dengan operasi Gauss serta solusi dari matriks solver
main	public static void	-	Menjalankan program class

3.10. Class SPL_GaussJordan

Atribut

- Class ini tidak memiliki atribut

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
Jordan	Public static void	Matrix **solver**	Menampilkan langkah langkah dengan operasi Gauss Jordan serta solusi dari matriks solver
main	Public static void	-	Menjalankan program class

3.11. Class IO

Atribut

- Class ini tidak memiliki atribut

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
inputMatrixFile	Public static Matrix	String **filename**	Menghasilkan matriks dari masukan file
outputOBEFile	Public static void	String **filename** String **array**	Membuat file hasil OBE (persamaan)
outputDeterminanFile	Public static void	String **filename** double det	Membuat file hasil determinan
outputInversFile	Public static void	String **filename** Matrix **matriks**	Membuat file hasil invers

3.12. Class Menu

Atribut

- Class ini tidak memiliki atribut

Method

Nama	**Tipe**	**Parameter**	**Deskripsi**
mainMenu	Public static void	-	Mengeluarkan tampilan main menu
subMenuSPL	Public static void	-	Mengeluarkan tampilan sub menu spl
subMenuDet	Public static void	-	Mengeluarkan tampilan sub menu determinan
subMenuInverse	Public static void	-	Mengeluarkan tampilan sub menu matriks balikan
main	Public static void	-	Menjalankan program matriks kalkulator

3.13. Class main

Atribut

- Class ini tidak memiliki atribut

Method

- Class ini tidak memiliki Method

BAB IV
Eksperimen

4.1. Tentukan Solusi SPL $Ax = B$

<table>
<tr>
<td>

4.1.a

Matriks Augmented :

$$\begin{bmatrix} 1 & 1 & -1 & -1 & | & 1 \\ 2 & 5 & -7 & -5 & | & -2 \\ 2 & -1 & 1 & 3 & | & 4 \\ 5 & 2 & -4 & 2 & | & 6 \end{bmatrix}$$

</td>
<td>

```
R2 -> R2 - 2.00*R1
1.00 1.00 -1.00 -1.00 1.00
0.00 3.00 -5.00 -3.00 -4.00
2.00 -1.00 1.00 3.00 4.00
5.00 2.00 -4.00 2.00 6.00

R3 -> R3 - 2.00*R1
1.00 1.00 -1.00 -1.00 1.00
0.00 3.00 -5.00 -3.00 -4.00
0.00 -3.00 3.00 5.00 2.00
5.00 2.00 -4.00 2.00 6.00

R4 -> R4 - 5.00*R1
1.00 1.00 -1.00 -1.00 1.00
0.00 3.00 -5.00 -3.00 -4.00
0.00 -3.00 3.00 5.00 2.00
0.00 -3.00 1.00 7.00 1.00

R2 -> R2/3.00
1.00 1.00 -1.00 -1.00 1.00
0.00 1.00 -1.67 -1.00 -1.33
0.00 -3.00 3.00 5.00 2.00
0.00 -3.00 1.00 7.00 1.00

R3 -> R3 + 3.00*R2
1.00 1.00 -1.00 -1.00 1.00
0.00 1.00 -1.67 -1.00 -1.33
0.00 0.00 -2.00 2.00 -2.00
0.00 -3.00 1.00 7.00 1.00
```

</td>
</tr>
</table>

```
R3 -> -R3/2.00
1.00 1.00 -1.00 -1.00 1.00
0.00 1.00 -1.67 -1.00 -1.33
0.00 0.00 1.00 -1.00 1.00
0.00 0.00 -4.00 4.00 -3.00

R4 -> R4 + 4.00*R3
1.00 1.00 -1.00 -1.00 1.00
0.00 1.00 -1.67 -1.00 -1.33
0.00 0.00 1.00 -1.00 1.00
0.00 0.00 0.00 0.00 1.00

Solusi tidak ada
```

4.1.b

Matriks Augmented :

$$\begin{bmatrix} 1 & -1 & 0 & 0 & 1 & | & 3 \\ 1 & 1 & 0 & -3 & 0 & | & 6 \\ 2 & -1 & 0 & 1 & -1 & | & 5 \\ -1 & 2 & 0 & -2 & -1 & | & -1 \end{bmatrix}$$

```
R2 -> R2 - R1
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 2.00 0.00 -3.00 -1.00 3.00
2.00 -1.00 0.00 1.00 -1.00 5.00
-1.00 2.00 0.00 -2.00 -1.00 -1.00

R3 -> R3 - 2.00*R1
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 2.00 0.00 -3.00 -1.00 3.00
0.00 1.00 0.00 1.00 -3.00 -1.00
-1.00 2.00 0.00 -2.00 -1.00 -1.00

R4 -> R4 + 1.00*R1
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 2.00 0.00 -3.00 -1.00 3.00
0.00 1.00 0.00 1.00 -3.00 -1.00
0.00 1.00 0.00 -2.00 0.00 2.00

R2 -> R2/2.00
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 1.00 0.00 -1.50 -0.50 1.50
0.00 1.00 0.00 1.00 -3.00 -1.00
0.00 1.00 0.00 -2.00 0.00 2.00
```

```
R3 -> R3 - R2
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 1.00 0.00 -1.50 -0.50 1.50
0.00 0.00 0.00 2.50 -2.50 -2.50
0.00 1.00 0.00 -2.00 0.00 2.00

R4 -> R4 - R2
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 1.00 0.00 -1.50 -0.50 1.50
0.00 0.00 0.00 2.50 -2.50 -2.50
0.00 0.00 0.00 -0.50 0.50 0.50

R3 -> R3/2.50
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 1.00 0.00 -1.50 -0.50 1.50
0.00 0.00 0.00 1.00 -1.00 -1.00
0.00 0.00 0.00 -0.50 0.50 0.50

R4 -> R4 + 0.50*R3
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 1.00 0.00 -1.50 -0.50 1.50
0.00 0.00 0.00 1.00 -1.00 -1.00
0.00 0.00 0.00 0.00 0.00 0.00

R2 -> R2 + 1.50*R3
1.00 -1.00 0.00 0.00 1.00 3.00
0.00 1.00 0.00 0.00 -2.00 0.00
0.00 0.00 0.00 1.00 -1.00 -1.00
0.00 0.00 0.00 0.00 0.00 0.00
```

```
R1 -> R1 + R2
1.00 0.00 0.00 0.00 -1.00 3.00
0.00 1.00 0.00 0.00 -2.00 0.00
0.00 0.00 0.00 1.00 -1.00 -1.00
0.00 0.00 0.00 0.00 0.00 0.00

Solusi dari persamaan diatas adalah:
X1 = X5 + 3.00
X2 = 2.00*X5
X3 = X3
X4 = X5 - 1.00
X5 = X5
```

4.1.c
Matriks Augmented :

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

```
Tukar matriks indeks ke-2 dengan indeks ke-3
0.00 1.00 0.00 0.00 1.00 0.00 2.00
0.00 1.00 0.00 0.00 0.00 1.00 1.00
0.00 0.00 0.00 1.00 1.00 0.00 -1.00

R2 -> R2 - R1
0.00 1.00 0.00 0.00 1.00 0.00 2.00
0.00 0.00 0.00 0.00 -1.00 1.00 -1.00
0.00 0.00 0.00 1.00 1.00 0.00 -1.00

Tukar matriks indeks ke-2 dengan indeks ke-3
0.00 1.00 0.00 0.00 1.00 0.00 2.00
0.00 0.00 0.00 1.00 1.00 0.00 -1.00
0.00 0.00 0.00 0.00 -1.00 1.00 -1.00

R3 -> -R3/1.00
0.00 1.00 0.00 0.00 1.00 0.00 2.00
0.00 0.00 0.00 1.00 1.00 0.00 -1.00
0.00 0.00 0.00 0.00 1.00 -1.00 1.00

R2 -> R2 - 1.00*R3
0.00 1.00 0.00 0.00 1.00 0.00 2.00
0.00 0.00 0.00 1.00 0.00 1.00 -2.00
0.00 0.00 0.00 0.00 1.00 -1.00 1.00

R1 -> R1 - 1.00*R3
0.00 1.00 0.00 0.00 0.00 1.00 1.00
0.00 0.00 0.00 1.00 0.00 1.00 -2.00
0.00 0.00 0.00 0.00 1.00 -1.00 1.00
```

```
Solusi dari persamaan diatas adalah:
X1 = X1
X2 = -X6 + 1.00
X3 = X3
X4 = -X6 - 2.00
X5 = X6 + 1.00
X6 = X6
```

4.1.d.1
Matriks Augmented :

$$\begin{bmatrix} 1 & 1/2 & 1/3 & .. & 1/6 & | & 1 \\ 1/2 & 1/3 & 1/4 & .. & 1/7 & | & 0 \\ 1/3 & 1/4 & 1/5 & .. & 1/8 & | & 0 \\ : & : & : & .. & : & | & : \\ 1/6 & 1/7 & 1/8 & .. & 1/11 & | & 0 \end{bmatrix}$$

```
X1 = 36.000
X2 = - 630.000
X3 = 3360.000
X4 = - 7560.000
X5 = 7560.000
X6 = - 2772.000
```

4.1.d.2

Matriks Augmented :

$$\left[\begin{array}{ccccc|c} 1 & 1/2 & 1/3 & .. & 1/10 & 1 \\ 1/2 & 1/3 & 1/4 & .. & 1/11 & 0 \\ 1/3 & 1/4 & 1/5 & .. & 1/12 & 0 \\ \vdots & \vdots & \vdots & .. & \vdots & \vdots \\ 1/10 & 1/11 & 1/12 & .. & 1/19 & 0 \end{array}\right]$$

```
X1 = 99.998
X2 = - 4949.798
X3 = 79195.672
X4 = - 600560.529
X5 = 2522331.254
X6 = - 6305780.063
X7 = 9608745.465
X8 = - 8750772.982
X9 = 4375365.278
X10 = - 923684.294
```

4.2. SPL Berbentuk Matriks *Augmented Matrix*

4.2.a

Matriks Augmented :

$$\left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -1 \\ 2 & 1 & -2 & -2 & -2 \\ -1 & 2 & -4 & 1 & 1 \\ 3 & 0 & 0 & -3 & -3 \end{array}\right]$$

```
R2 -> R2 - 2.00*R1
1.00 -1.00 2.00 -1.00 -1.00
0.00 3.00 -6.00 0.00 0.00
-1.00 2.00 -4.00 1.00 1.00
3.00 0.00 0.00 -3.00 -3.00

R3 -> R3 + 1.00*R1
1.00 -1.00 2.00 -1.00 -1.00
0.00 3.00 -6.00 0.00 0.00
0.00 1.00 -2.00 0.00 0.00
3.00 0.00 0.00 -3.00 -3.00

R4 -> R4 - 3.00*R1
1.00 -1.00 2.00 -1.00 -1.00
0.00 3.00 -6.00 0.00 0.00
0.00 1.00 -2.00 0.00 0.00
0.00 3.00 -6.00 0.00 0.00

R2 -> R2/3.00
1.00 -1.00 2.00 -1.00 -1.00
0.00 1.00 -2.00 0.00 0.00
0.00 1.00 -2.00 0.00 0.00
0.00 3.00 -6.00 0.00 0.00

R3 -> R3 - R2
1.00 -1.00 2.00 -1.00 -1.00
0.00 1.00 -2.00 0.00 0.00
0.00 0.00 0.00 0.00 0.00
0.00 3.00 -6.00 0.00 0.00
```

	R4 -> R4 - 3.00*R2 1.00 -1.00 2.00 -1.00 -1.00 0.00 1.00 -2.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 R1 -> R1 + R2 1.00 0.00 0.00 -1.00 -1.00 0.00 1.00 -2.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 Solusi dari persamaan diatas adalah: X1 = X4 - 1.00 X2 = 2.00*X3 X3 = X3 X4 = X4
4.2.b Matriks Augmented : $\begin{bmatrix} 2 & 0 & 8 & 0 & \vert & 8 \\ 0 & 1 & 0 & 4 & \vert & 6 \\ -4 & 0 & 6 & 0 & \vert & 6 \\ 0 & -2 & 0 & 3 & \vert & -1 \\ 2 & 0 & -4 & 0 & \vert & -4 \\ 0 & 1 & 0 & -2 & \vert & 0 \end{bmatrix}$	Solusi dari persamaan diatas adalah: X1 = 0.00 X2 = 2.00 X3 = 1.00 X4 = 1.00

4.3. SPL Berbentuk Normal

4.3.a Sistem Persamaan Linear : $8x_1 + x_2 + 3x_3 + 2x_4 = 0$ $2x_1 + 9x_2 - x_3 - 2x_4 = 1$ $x_1 + 3x_2 + 2x_3 - x_4 = 2$ $x_1 + 6x_3 + 4x_4 = 3$	Solusi dari persamaan diatas adalah: X1 = - 0.22 X2 = 0.18 X3 = 0.71 X4 = - 0.26

4.3.b
Sistem Persamaan Linear :

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

Tidak ada solusi

```
1.00 1.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00 8.00
0.00 1.00 -4.20 5.51 0.32 0.00 0.32 0.00 0.00 -8.10
0.00 0.00 1.00 -1.25 -0.06 0.00 -0.25 0.00 0.00 1.94
0.00 0.00 0.00 1.00 0.06 1.40 1.25 0.00 0.08 16.80
0.00 0.00 0.00 0.00 1.00 1.34 0.19 0.26 1.03 24.16
0.00 0.00 0.00 0.00 0.00 1.00 0.32 -0.76 0.01 5.11
0.00 0.00 0.00 0.00 0.00 0.00 1.00 -5.51 -2.55 -43.85
0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00 0.31 7.57
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00 5.00
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.51 -2.46
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 -0.00
```

4.4. Studi Kasus Interpolasi Polinom

4.4.a

x	0.4	0.7	0.11	0.14	0.17	0.2	0.23
$f(x)$	0.043	0.005	0. 058	0.072	0.1	0.13	0.147

Gunakan Tabel di atas untuk mencari nilai-nilai default berikut:

a. $x = 0.2$
b. $x = 0.55$
c. $x = 0.85$
d. $x = 1.28$

4.4.a.a

```
Interpolasi Polinom
Input file (y/n) : n
Masukkan Input N: 7
Masukkan x y: 0.4 0.043
Masukkan x y: 0.7 0.005
Masukkan x y: 0.11 0.058
Masukkan x y: 0.14 0.072
Masukkan x y: 0.17 0.1
Masukkan x y: 0.2 0.13
Masukkan x y: 0.23 0.147
Nilai yang ingin ditaksir:
0.2
P(x) = -0.1845590191852713 + 10.276383990257301 X - 163.91566262198512 X^2 + 122
0.8548907126942 X^3 - 4346.313951121293 X^4 + 7102.399162995146 X^5 - 4212.43453
20746315 X^6
P(0.20) = 0.13
```

4.4.a.b	`P(x) = -0.1845590191852713 + 10.276383990257301 X - 163.91566262198512 X^2 + 122`
	`0.8548907126942 X^3 - 4346.313951121293 X^4 + 7102.399162995146 X^5 - 4212.43453`
	`20746315 X^6`
	`P(0.55) = 2.14`

4.4.a.c	`P(x) = -0.1845590191852713 + 10.276383990257301 X - 163.91566262198512 X^2 +`
	`0.8548907126942 X^3 - 4346.313951121293 X^4 + 7102.399162995146 X^5 - 4212.4`
	`20746315 X^6`
	`P(0.85) = -66.27`

4.4.a.d	`P(x) = -0.1845590191852713 + 10.276383990257301 X - 163.91566262198512 X^2 +`
	`0.8548907126942 X^3 - 4346.313951121293 X^4 + 7102.399162995146 X^5 - 4212.4`
	`20746315 X^6`
	`P(1.23) = -2503.32`

4.4.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	12624
30/06/2022	7	21807
08/07/2022	7.258	38391
14/07/2022	7.451	54517
26/07/2022	7.839	28228
05/08/2022	8.161	35764

15/08/2022	8.484	20813
22/08/2022	8.709	12408
31/08/2022	9	10534

Setelah dilakukan perhitungan diperoleh persamaan

```
P(x) = 7.191043175045852E12 + -9.35161531589144E12 X + 5.336587667141117E12 X^2
- 1.757526932593959E12 X^3 + 3.6868913056006055E11 X^4 - 5.114965157579532E10 X^
5 + 4.697327200671469E9 X^6 - 2.755580954627556E8 X^7 + 9375523.81015 X^8 - 1410
31.71617072617 X^9
```

Tanggal	**Hasil interpolasi polinom**
16/07/2022	`P(x) = 7.191043175045852E12 + -9.35161531589144E12 X + 5.336587667141117E12 X^2 - 1.757526932593959E12 X^3 + 3.6868913056006055E11 X^4 - 5.114965157579532E10 X^5 + 4.697327200671469E9 X^6 - 2.755580954627556E8 X^7 + 9375523.81015 X^8 - 141031.71617072617 X^9` `P(7.52) = 53375.36`
10/08/2022	`P(x) = 7.191043175045852E12 + -9.35161531589144E12 X + 5.336587667141117E12 X^2 - 1.757526932593959E12 X^3 + 3.6868913056006055E11 X^4 - 5.114965157579532E10 X^5 + 4.697327200671469E9 X^6 - 2.755580954627556E8 X^7 + 9375523.81015 X^8 - 141031.71617072617 X^9` `P(8.32) = 36439.88`
05/09/2022	`P(x) = 7.191043175045852E12 + -9.35161531589144E12 X + 5.336587667141117E12 X^2 - 1.757526932593959E12 X^3 + 3.6868913056006055E11 X^4 - 5.114965157579532E10 X^5 + 4.697327200671469E9 X^6 - 2.755580954627556E8 X^7 + 9375523.81015 X^8 - 141031.71617072617 X^9` `P(9.17) = -667748.82`
3/10/2022	`P(x) = 7.191043175045852E12 + -9.35161531589144E12 X + 5.336587667141117E12 X^2 - 1.757526932593959E12 X^3 + 3.6868913056006055E11 X^4 - 5.114965157579532E10 X^5 + 4.697327200671469E9 X^6 - 2.755580954627556E8 X^7 + 9375523.81015 X^8 - 141031.71617072617 X^9` `P(10.10) = -337121866.88`

4.4.c

Sederhanakan fungsi

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

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

Maka diperoleh

x	$f(x)$
0	0
0.4	0.41888423
0.8	0.507157969

1.2	0.583685661
2	0.57665153

```
Pilihan : 4
Interpolasi Polinom
Input file (y/n) : n
Masukkan Input N: 6
Masukkan x y: 0 0
Masukkan x y: 0.4 0.41888423
Masukkan x y: 0.8 0.507157969
Masukkan x y: 1.2 0.560924675
Masukkan x y: 1.6 0.583685661
Masukkan x y: 2 0.57665153
Nilai yang ingin ditaksir:
1.3
P(x) = 2.035258628541669 X - 3.55268368046876 X^2 + 3.23711376757814 X^3 - 1.4212646728515719 X^4 + 0.23625605061849164 X^5
P(1.30) = 0.57
```

Sehingga diperoleh persamaan

$f(x) = 2.03525x - 3.552683x^2 + 3.237113767x^3 - 1.4212646x^4 + 0.236256x^5$

4.5. Studi Kasus Interpolasi Bicubic

Diberikan matriks input:

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

Tentukan nilai:

a. $f(0, 0)$
b. $f(0.5, 0.5)$
c. $f(0.25, 0.75)$
d. $f(0.1, 0.9)$

4.5.a	`Masukkan nama file (filename.txt): bicubic.txt` `f(0.0,0.0) = 161.0`
4.5.b	`Masukkan nama file (filename.txt): bicubic.txt` `f(0.5,0.5) = 97.73`
4.5.c	`Masukkan nama file (filename.txt): bicubic.txt` `f(0.25,0.75) = 105.51`
4.5.d	`Masukkan nama file (filename.txt): bicubic.txt` `f(0.1,0.9) = 104.23`

4.6. Studi Kasus Regresi Linear Berganda

Diberikan data sebagai berikut

x_1	x_2	x_3	y
72.4	76.3	29.18	0.9

41.6	70.3	29.35	0.91
34.3	77.1	29.24	0.96
35.1	68	29.27	0.89
10.7	79	29.78	1
12.9	67.4	29.39	1.1
8.3	66.8	29.69	1.15
20.1	76.9	29.48	1.03
72.2	77.7	29.09	0.77
24	67.7	29.6	1.07
23.2	76.8	29.38	1.07
47.4	86.6	29.35	0.94
31.5	76.9	29.63	1.1
10.6	86.3	29.56	1.1
11.2	86	29.48	1.1
73.3	76.3	29.4	0.91
75.4	77.9	29.28	0.87
96.6	78.7	29.29	0.78
107.4	86.8	29.03	0.82
54.9	70.9	29.37	0.95

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. Dengan menggunakan persamaan RLB diperoleh

```
Diperoleh hasil RLB Ganda
1.00 0.00 0.00 0.00 -3.50
0.00 1.00 0.00 0.00 -0.00
0.00 0.00 1.00 0.00 0.00
0.00 0.00 0.00 1.00 0.15
Y = -3.505 - 0.003*X1 + 0.001*X2 + 0.154*X3
Hasil taksiran nilai y dari regresi linear adalah : 0.938
```

4.7. Image Scaling

Input	Output
https://github.com/farhanfahrezy/Tubes-1-Algeo-SPL/blob/main/test/output.jpg	https://github.com/farhanfahrezy/Tubes-1-Algeo-SPL/blob/main/hasil/output.png

BAB V
Kesimpulan Saran dan Refleksi

5.1. Kesimpulan

Pada tugas ini telah dibuat library dalam bentuk .jar yang secara khusus digunakan untuk mengolah matriks dengan operasi-operasi tertentu, seperti mencari solusi persamaan linear, mencari determinan matriks, mencari invers matriks dan pengaplikasiannya.

Penyelesaian dari sebuah SPL terbagi 3, yaitu tidak ada solusi, banyak (tidak berhingga), atau hanya satu (unik/tunggal). Dari eksperimen yang telah dilakukan, salah satu persamaan yang tidak memiliki solusi adalah soal 4.1.a karena setelah dilakukan eliminasi Gauss, baris paling bawah memiliki nilai variabel nol tetapi nilai persamaannya tidak nol.

Persamaan yang memiliki banyak solusi bisa dilihat pada kasus 4.2.a. Persamaan yang hanya memiliki satu solusi bisa dilihat pada kasus 4.1.d yaitu kasus matriks hilbert.

Kendala yang dialami pada Tugas Besar kali ini adalah kurangnya keakuratan penghitungan, namun keakuratan pada program yang telah dibuat sudah mendekati nilai yang sebenarnya, hal ini disebabkan oleh penggunaan variable double yang menyebabkan perhitungan menjadi kurang akurat.

5.2. Saran

Pada program yang dibuat pada tugas ini, belum tercapai modularisasi yang sempurna. Kurangnya keakuratan dalam penghitungan menyebabkan program menjadi kurang sempurna. Jika diberikan kerangka program yang terstruktur bisa membuat waktu pengerjaan dari tugas ini menjadi lebih singkat dan pembagian tugas juga jadi lebih mudah.

5.3. Refleksi

Melalui tugas besar pertama ini, kami merasa bahwa di jurusan Informatika tidaklah mudah dan lebih menantang dari masa TPB, dimana tugas besar diberikan dalam kurun waktu yang relatif singkat, dan tentunya dalam tugas besar pertama ini kami dapat belajar bukan hanya dalam bidang pemrograman namun belajar mengkoordinasi waktu, melakukan pembagian yang pas, koordinasi dengan sesama anggota dan masih banyak hal lainnya.

Referensi

https://www.cuemath.com/algebra/inverse-of-a-matrix/ (diakses pada 01/10/2022)

https://www.kelaspintar.id/blog/tips-pintar/adjoin-matriks-11543/#:~:text=Kofaktor%20merupakan%20hasil%20perkalian%20minor,dikenali%20melalui%20lambangnya%2C%20yaitu%20Cij (diakses pada 01/10/2022)

https://id.wikipedia.org/wiki/Determinan (diakses pada 01/10/2022)

https://id.wikipedia.org/wiki/Kaidah_Cramer (diakses pada 01/10/2022)

Link Repository :
https://github.com/farhanfahrezy/Algeo01-21080