TUGAS BESAR 1 IF2123

ALJABAR LINEAR DAN GEOMETRI

2022/2023

Diajukan sebagai salah satu tugas mata kuliah Aljabar Linear dan Geometri pada Semester I Tahun Akademik 2022/2023

oleh

Henry Anand Septian Radityo	**13521004**
Christophorus Dharma Winata	**13521009**
Fahrian Afdholi	**13521031**

BAB I
DESKRIPSI MASALAH

Program dapat menerima masukan (input) baik dari keyboard maupun membaca masukan dari file text. Untuk SPL, masukan dari keyboard adalah m, n, koefisien aij , dan bi . 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
```

Untuk persoalan menghitung determinan dan matriks balikan, masukan dari keyboard adalah n dan koefisien aij . 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
```

Untuk persoalan interpolasi, masukannya jika dari keyboard adalah n, (x0 , y0), (x1 , y1), ..., (xn , yn), dan nilai x yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung. Misalnya jika titik-titik datanya adalah (8.0, 2.0794), (9.0, 2.1972), dan (9.5, 2.2513), maka di dalam file text ditulis sebagai berikut:

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

Untuk persoalan regresi, masukannya jika dari keyboard adalah n (jumlah peubah x), m (jumlah sampel), semua nilai-nilai x1i , x2i , ..., xni , nilai yi , dan nilai-nilai xk yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung.

Untuk persoalan SPL, luaran (output) program adalah solusi SPL. Jika solusinya tunggal, tuliskan nilainya. Jika solusinya tidak ada, tuliskan solusi tidak ada, jika solusinya banyak, maka tuliskan solusinya dalam bentuk parametrik (misalnya x4 = -2, x3 = 2s – t, x2 = s, dan x1 = t.)

Untuk persoalan determinan dan matriks balikan, maka luarannya sesuai dengan persoalan masing-masing

```
-3 7 8.3 11 -4
3 4.5 2.8 10 12
0.5 -10 -9 12 0
0.1 0.2
```

Untuk persoalan polinom interpolasi dan regresi, luarannya adalah persamaan polinom/regresi dan taksiran nilai fungsi pada x yang diberikan. Contoh luaran untuk interpolasi adalah $f(x) = -0.0064x^2 + 0.2266x + 0.6762$, $f(5) = ...$ dan untuk regresi adalah $f(x) = -9.5872 + 1.0732x_1$, $f(x_k) =$...

Untuk persoalan interpolasi bicubic, masukan dari file text (.txt) yang berisi matriks berukuran 4x4 yang berisi nilai f(i,j) dengan i dan j adalah indeks matriks diikuti dengan nilai a dan b untuk mencari nilai f(a,b). misalnya jika nilai dari f(-1,-1), f(-1,0), f(-1,1), f(-1,2),f(0,-1), f(0,0), f(0,1), f(0,2), f(1,-1), f(1,0), f(1,1), f(1,2), f(2,-1), f(2,0), f(2,1), f(2,2) 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).
masukannya adalah matriks 4 x 4, diikuti oleh nilai a dan b, maka luarannya adalah nilai f(a,b).

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

Bahasa program yang digunakan adalah Java.

Program tidak harus berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas Eclipse misalnya).

Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

```
MENU
1. Sistem Persamaaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
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 II
TEORI SINGKAT

2.1 Metode Eliminasi Gauss

Metode ini merupakan salah satu metode untuk mencari solusi dari Sistem Persamaan Linear yang diubah menjadi matriks augmented, lalu dilakukan operasi baris elementer sampai terbentuk sebuah matriks eselon baris dengan satu utama di setiap barisnya, kecuali pada baris yang seluruh elemennya bernilai nol.

Setelah didapatkan matriks eselon baris, kemudian dilakukan substitusi mundur atau metode penyelesaian gauss jordan sehingga didapatkan sebuah solusi dari sebuah Sistem Persamaan Linear.

2.2 Metode Eliminasi Gauss - Jordan

Metode Gauss-Jordan merupakan metode lanjutan dari metode eliminasi gauss untuk mendapatkan solusi dari SPL. Metode ini terdiri dari dua fase yaitu fase maju yang merupakan metode eliminasi gauss, dan fase kedua yaitu fase mundur dimana pada fase ini dibentuk matriks eselon baris tereduksi atau di atas dari satu utama terdapat nilai nol.

2.3 Determinan

Determinan merupakan pemetaan dengan domain berupa bujur sangkar atau matriks persegi, determinan memiliki beberapa fungsi seperti menemukan invers dari suatu matriks, menentukan solusi dari sistem persamaan linier dengan metode cramer, dan memeriksa basis di suatu ruang vektor.

Untuk mencari sebuah determinan dari suatu matriks dapat dicari dengan berbagai cara yaitu dengan cara kofaktor dan dengan cara operasi baris elementer. Cara menghitung determinan matriks dengan cara kofaktor adalah dengan mengubah suatu matriks menjadi matriks segitiga atas atau segitiga bawah, yaitu matriks yang hanya memiliki nilai selain nol di salah satu sisi yang dibagi oleh diagonalnya. Setelah terbentuk matriks segitiga atas atau segitiga bawah selanjutnya determinan dari matriks dapat dihitung dengan rumus

$$det(A) = \frac{(-1)^p \times a'_{11} \times a'_{22} \times ... \times a'_{nn}}{k_1 \times k_2 \times k_3 \times ... \times k_n}$$

Dengan a merupakan elemen dari matriks pada diagonal, p adalah berapa kali operasi baris elementer dilakukan, sedangkan k adalah berapa kali dilakukan perkalian baris terhadap suatu konstanta.

Pencarian determinan dari sebuah matriks dengan metode kofaktor adalah dengan mencari sebuah sub matriks atau matriks baru yang elemennya berupa minor dari matriks asal. $Minor_{12}$dari matriks A adalah sebuah matriks A yang baris 1 dan kolom 2 nya tidak diikutsertakan. Selanjutnya untuk membuat suatu matriks kofaktor diperlukan juga sebuah penentuan tanda positif dan negatif, jika suatu baris dan kolom yang dijumlahkan bernilai ganjil maka tanda yang dihasilkan adalah negatif sedangkan apabila penjumlahan index baris dan kolomnya genap akan dihasilkan hasil atau tanda positif. Selanjutnya determinan dapat ditentukan dengan salah satu baris dari matriks kofaktor yang telah ditentukan sebelumnya.

2.4 Matriks Balikan

Matriks balikan adalah matriks yang apabila dikalikan dengan matriks itu sendiri akan dihasilkan sebuah matriks identitas, hal ini juga biasanya disebut sebagai matriks komplemen. Matriks balikan dapat dicari dengan dua cara yaitu dengan metode adjoin yang melibatkan sebuah determinan dan juga matriks adjoin.

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

Matriks adjoin A dapat dicari dengan melakukan transpose terhadap matriks kofaktor dari A. Matriks balikan juga dapat dicari dengan menggunakan metode OBE dengan mengaugmentedkan suatu matriks dengan suatu matriks identitas. [A|I] lalu dilakukan OBE hingga ditemukan bentuk $[I|A^{-1}]$ apabila sisi kiri sudah berubah menjadi matriks identitas maka sisi kanan adalah invers dari matriks tersebut.

2.5 Matriks Kofaktor

Matriks kofaktor adalah sebuah matriks yang terdiri dari minor - minor matriks asalnya, $Minor_{12}$ adalah sebuah submatriks dimana baris 1 dan kolom 2 tidak diikutkan pada matriks. Selanjutnya tanda ditentukan dari penjumlahan indeks baris dan juga indeks kolom, apabila hasilnya genap maka tanda positif apabila ganjil maka negatif.

2.6 Matriks Adjoin

Matriks adjoin adalah matriks yang dapat dibentuk dengan melakukan transpose pada matriks kofaktor.

2.7 Kaidah Cramer

Kaidah Cramer adalah salah satu cara untuk membuat solusi dari sistem persamaan linier, kaidah cramer bekerja dengan membuat suatu matriks dari SPL namun kolom terakhir dihilangkan dan dipisahkan menjadi sebuah matriks baru. Sebuah matriks A apabila ingin dicari sebuah solusi dari persamaan barisnya adalah dengan menentukan determinan dari A dan juga determinan dari A_1hingga A_n. A_1 adalah sebuah matriks A yang kolom pertamanya diganti dengan kolom terakhir atau hasil dari sebuah SPL. Lalu solusi dari tiap variabelnya dapat dicari dengan menggunakan rumus $\frac{A_n}{A}$.

2.8 Interpolasi Polinom

Interpolasi polinom adalah sebuah metode untuk menentukan adanya fungsi P, yaitu fungsi yang melewati semua titik yang diberikan di awal. Untuk menentukan fungsi P dibutuhkan suatu matriks yang mewakili semua titik yang ada yang dapat dibuat dengan

$$a_0 + a_1x_0 + a_2{x_0}^2 + \dots + a_n{x_0}^n = y_0$$
$$a_0 + a_1x_1 + a_2{x_1}^2 + \dots + a_n{x_1}^n = y_1$$
$$\dots$$
$$a_0 + a_1x_n + a_2{x_n}^2 + \dots + a_n{x_n}^n = y_n$$

Setelah dibuat matriksnya dapat dicari solusi untuk a dengan melakukan metode eliminasi gauss dan substitusi mundur. Setelah dibuat fungsi polinom yang melalui berbagai titik maka P(n) dapat langsung dicari dengan substitusi.

2.9 Interpolasi Bicubic

Interpolasi bicubic merupakan salah satu interpolasi yang sering digunakan dalam mencari suatu nilai diantara nilai tertentu. Interpolasi ini bekerja dengan melibatkan tiga matriks yaitu matriks a sebagai matriks yang perlu dicari, matriks y yaitu matriks yang sudah dimiliki, dan matriks x yaitu matriks yang berisi kaidah

$$f(x,y) = \sum_{j=0}^{3}\sum_{i=0}^{3} a_{ij}x^i y^j$$
$$x = -1, 0, 1, 2$$

Setelah ketiga matriks tersebut didapatkan, maka matriks a dapat dihitung dengan menggunakan invers.

$$a = X^{-1}y$$

Setelah mendapatkan matriks a, maka dapat dibuat suatu persamaan yang mampu menginterpolasi beberapa daerah.

2.10 Regresi Linier Berganda

Fungsi regresi adalah fungsi pada matematika yang menjelaskan suatu relasi satu atau lebih data variabel independen dengan suatu nilai yang dependen terhadap nilai-nilai independen tersebut. Fungsi regresi linear menghubungkan dari variabel-variabel tersebut dan mengasumsi kedua variabel dapat didekati oleh suatu persamaan garis lurus. Aplikasi dari fungsi ini adalah untuk memprediksi suatu nilai dependen dari suatu data-data independen berdasarkan data-data yang diproses dalam regresi.

Dalam komputasi, kalkulasi fungsi regresi linear yang berganda dapat diterapkan karena untuk menghitung regresi akan lebih mudah jika data-data dibentuk menjadi suatu matriks lalu memprosesnya. Adapun cara penghitungannya adalah sebagai berikut.

Data-data independen disusun menjadi matriks X dan data yang dependen disusun menjadi matriks y sesuai dengan matriks X. Lalu matriks-matriks tersebut dihitung sebagai berikut:

$$X'X = \begin{bmatrix} n & \sum_{i=1}^{n} x_{1i} & \cdot\cdot & \sum_{i=1}^{n} x_{2i} & \sum_{i=1}^{n} x_{ki} \\ \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{1i}^2 & \cdot\cdot & \sum_{i=1}^{n} x_{1i}x_{2i} & \sum_{i=1}^{n} x_{1i}x_{ki} \\ \vdots & \vdots & \cdot\cdot & \vdots & \vdots \\ \sum_{i=1}^{n} x_{ki} & \sum_{i=1}^{n} x_{ki}x_{1i} & \cdot\cdot & \sum_{i=1}^{n} x_{ki}x_{2i} & \sum_{i=1}^{n} x_{ki}^2 \end{bmatrix} \quad X'y = \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_1 \\ \sum_{i=1}^{n} x_{1i}y_i \\ \vdots \\ \sum_{i=1}^{n} x_{ki}y_i \end{bmatrix}$$

Dari data tersebut dihitung sebagai

$$(X'X)\beta = X'y$$

Dan bisa didapat nilai-nilai β dengan eliminasi Gauss.

BAB III
IMPLEMENTASI PUSTAKA

3.1 Cramer.java

Method	Deskripsi
public static double[] cramer(double[][] A)	Menghitung solusi SPL dan mengembalikan nilai solusi SPL dalam bentuk array.
private static double[][] InsertBtoA (double[][] arr, double[][] b,int col)	Mengganti kolom pada index “col” dengan matriks B.

3.2 DeterminanKofaktor.java

Method	Deskripsi
public static double determinan(double[][] arr)	Menghitung determinan sebuah matriks dengan menggunakan metode kofaktor.

3.3 DeterminanOBE.java

Pada class ini didefinisikan beberapa atribut berupa :

- sign berfungsi sebagai konstanta minor entri
- det berfungsi sebagai nilai determinan

Method	Deskripsi
private static boolean HasAll0RowCol(double[][] m)	Mengembalikan nilai true apabila terdapat baris/kolom yang berisi 0 semua.
private static boolean All0rowcol(double[][] m, boolean isRow, int idx)	Fungsi pembantu HasAll0RowCol yang memeriksa setiap elemen pada suatu baris atau suatu kolom bernilai 0 semua. Nilai boolean isRow jika bernilai true akan memeriksa semua elemen pada baris idx, sedangkan jika bernilai false akan memeriksa kolom idx.
private static void kurangBasis(double[][] m, int row, double[] basis)	Prosedur yang mengurangi baris index row pada matriks m dengan sebuah baris basis
private static void tambahBasis(double[][] m, int row, double[] basis)	Prosedur yang menambah baris index row pada matriks m dengan sebuah baris basis
private static void printBasis(double[] basis)	Prosedur untuk menampilkan baris yang sekarang menjadi basis dalam eliminasi elemen matriks
private static void rowKaliConst(double[][] arr, int brs, double n)	Prosedur yang mengalikan baris index brs pada suatu matriks arr dengan konstanta n

private static void swapRow(double[][] arr, int brs1, int brs2)	Prosedur yang menukar baris pada matriks arr yang berada di index brs1 dengan baris index brs2
private static void obe(double[][] m)	Prosedur eliminasi elemen-elemen pada matriks sehingga bisa didapat matriks segitiga atas
public static double determinan(double[][] m)	Fungsi yang menggabungkan semua prosedur dan fungsi pada class untuk mendapatkan nilai determinan suatu matriks

3.4 Gauss.java

Pada class ini didefinisikan suatu object berupa :

- tools yang merupakan object dari Operator()

Method	Deskripsi
double[][] matrixGauss(double[][] m)	Mengembalikan sebuah matriks eselon baris dengan satu utama di setiap barisnya kecuali baris yang semua elemennya bernilai nol.

3.5 GaussJordan.java

Method	Deskripsi
public static double[][] matrixGaussJordan(double[][] m)	Mengembalikan sebuah matriks eselon baris tereduksi dengan satu utama di setiap barisnya dan angka di atas dan di bawah kecuali baris yang semua elemennya bernilai nol.

3.6 InterpolasiBicubic.java

Method	Deskripsi
public static double[][] makeMatriksX()	Mengembalikan matriks X dengan model $f(x, y) = \sum_{j=0}^{3}\sum_{i=0}^{3} a_{ij}x^i y^j$
public static double[] makeMatriksY(double[][] arr)	Mengembalikan sebuah array yang berisi semua elemen - elemen di dalam matriks arr.
public static double[] makematriksA(double[][] arr)	Mengembalikan sebuah array yang berisi nilai dari setiap elemen a.
public static double hitungFungsi(double[][] arr, double x, double y)	Mengembalikan sebuah nilai dari suatu fungsi yang telah dibuat.

3.7 InterpolasiPolinom.java

Method	Deskripsi

public static double[][] makePolinom(double[][] arr)	Mengembalikan sebuah array yang berisi SPL polinom dari titik - titik yang telah diberikan.
public static double[] makeFungsi(double[][] arr)	Membuat fungsi yang melalui semua titik input.
public static double hitungFungsi(double[][] arr, double x)	Mengembalikan nilai dari fungsi dari input x.

3.8 InverseAdjoin.java

Method	Deskripsi
public static double[][] inverseMatrixAdjoin(double[][] arr)	Mengembalikan sebuah matriks yang merupakan invers dari matriks arr menggunakan metode adjoin dan kofaktor.

3.9 InverseOBE.java

Method	Deskripsi
public static double[][] inverseMatrixOBE(double[][] m)	Mengembalikan sebuah matriks yang merupakan invers dari matriks arr menggunakan metode operasi baris elementer.

3.10 InverseSPL.java

Method	Deskripsi
public static double[] matrixInverseSPL(double[][] m)	Mengembalikan array yang berisi solusi SPL menggunakan metode invers matriks.

3.11 Operator.java

Method	Deskripsi
public static void printMatrix(double[][] m)	Mengembalikan nilai dari setiap elemen pada matriks ke layar.
public static void changeRow(double[][] m,int idxRow1)	Melakukan operasi pertukaran baris apabila nilai dari elemen di bawah idxRow1 tidak nol.
public static boolean isNotAllNol(double[][] m,int idxRow,int idxCol)	Mengembalikan nilai true apabila setiap kolom ke idxCol dari baris ke idxRow hingga akhir terdapat nilai nol.

public static double[][] bacaMatriks(int row, int col)	Membaca sebuah matriks dari keyboard dengan ukuran row x col.
public static double[][] sub_matriks(double[][] arr, int brs, int kol)	Mengembalikan suatu array dengan tidak memasukkan elemen baris ke brs dan kolom ke kol.
public static double[][] copyMatriks(double[][] arr, double[][] copy)	Mengembalikan nilai matriks yang sama dengan matriks arr.
public static void transpose(double[][] arr)	Melakukan operasi transpose pada matriks arr.
public static double[][] adjoin(double[][] arr)	Membuat adjoin dari matriks arr.
public static boolean isSquare(double[][] arr)	Mengembalikan nilai true apabila arr merupakan matriks persegi.
public static double[][] getMatrix(double[][] m,int idxRow1,int idxCol1,int idxRow2,int idxCol2)	Mengembalikan sebuah matriks yang berisi elemen - elemen dari idxRow1 sampai idxRow2 dan idxCol1 sampai idxCol2.
public static void printResultGaussJordan(double[][] m)	Mengeluarkan output dari nilai tiap x dari method gauss jordan
public static void printResultInverseSPL(double[] r)	Mengeluarkan output solusi dari SPL.
public static void createIdentity(double[][] m)	Mengubah matriks m menjadi sebuah matriks identitas.
public static double[][] mergeMatrix(double[][] m1,double[][] m2)	Mengembalikan matriks yang merupakan gabungan dari matriks m1 dan m2.
public static double[][] multiplyMatrix(double[][] m1, double[][] m2)	Melakukan proses perkalian matriks dari matriks m1 dan matriks m2
public static double[][] transposeNotSq(double[][] m)	Fungsi untuk transpose matriks non square

3.12 TxT.java

Method	Deskripsi
public static double[][] readMatrix(String path,int elementRow,int elementCol) throws IOException	membaca matrix yang terdapat pada txt di lokasi path dengan mengembalikan objek dengan format double[][]
public static boolean writeMatrix(String	menulis matrix dengan format double[][]

path,double[][] m)	dengan menghasilkan txt yang disimpan pada path
public static boolean writeInverseSPL(String path,double[][] m)	Mengembalikan output solusi dari SPL yang dihitung dengan metode invers matriks apabila determinan dari matriks SPL yang dibuat tidak bernilai nol.
public static boolean writeGaussSPL(String path,double[][] m)	Mengembalikan output dari SPL dengan metode Gauss.
public static boolean writeGaussJordanSPL(String path,double[][] m)	Mengembalikan output solusi dari SPL yang dihitung dengan metode Gauss Jordan.
public static boolean writeCramer(String path, double[][] m)	Mengembalikan solusi dari SPL yang dihitung dengan metode cramer.
public static boolean writeDeterminanKofaktor(String path,double[][] m)	Mengembalikan nilai dari determinan sebuah matriks dengan menggunakan metode kofaktor.
public static boolean writeDeterminanOBE(String path,double[][] m)	Mengembalikan nilai dari determinan sebuah matriks dengan menggunakan metode operasi baris elementer.
public static boolean writeInterpolasiBikubik(String path,double[][] m)	Mengembalikan output fungsi dari interpolasi bicubic.
public static boolean writeInterpolasiPolinom(String path, double[][] m)	Mengembalikan output solusi dari sebuah fungsi dari intepolasi polinom.

3.13 main.java

Method	Deskripsi
public static void main(String[] args)	

3.14 menu.java

Method	Deskripsi
public static int menu()	Mengeluarkan display dari menu dan menerima input menu.

public static int splDetInverse()	Mengeluarkan display dari cara penyelesaian SPL dan menerima input cara penyelesaian.

BAB IV
EKSPERIMEN

4.1 Solusi SPL

4.1.1 Studi Kasus 1

Dengan Kaidah Cramer

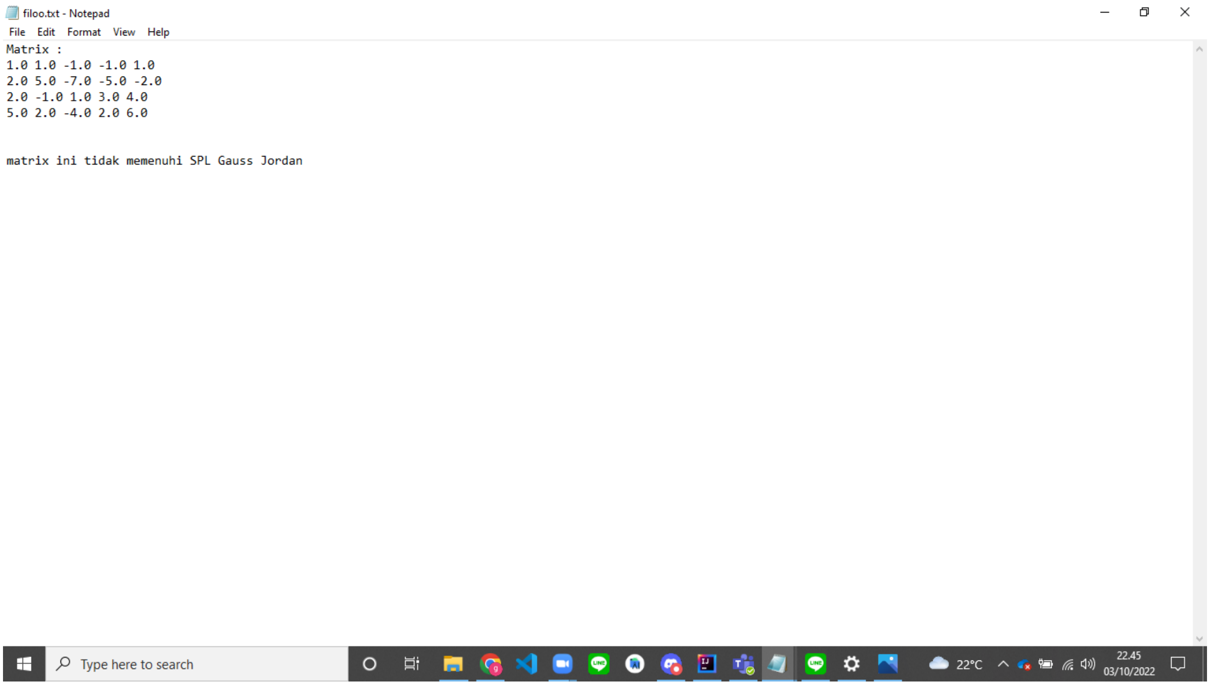
```
==============================================================SELAMAT DATANG KE DALAM PROGRAM KELOMPOK KAMI :)))==============================================================
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 1
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:4
masukkan jumlah kolom matrix:5
masukkan nama file txt:
..\test\Soal\StudiKasus1a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/Hasil/StudiKasus1aCramer.txt


==========================================CRAMER==========================================
Matrix :
1.0 1.0 -1.0 -1.0 1.0
2.0 5.0 -7.0 -5.0 -2.0
2.0 -1.0 1.0 3.0 4.0
5.0 2.0 -4.0 2.0 6.0
dengan metode cramer, persamaan ini memilki hasil:
Tidak bisa ditemukan hasil karena determinan matriks awal sama dengan nol
x1 = 0.0
x2 = 0.0
x3 = 0.0
x4 = 0.0
==========================================================================================


matrix berhasil diproses
```

Dengan kaidah gauss

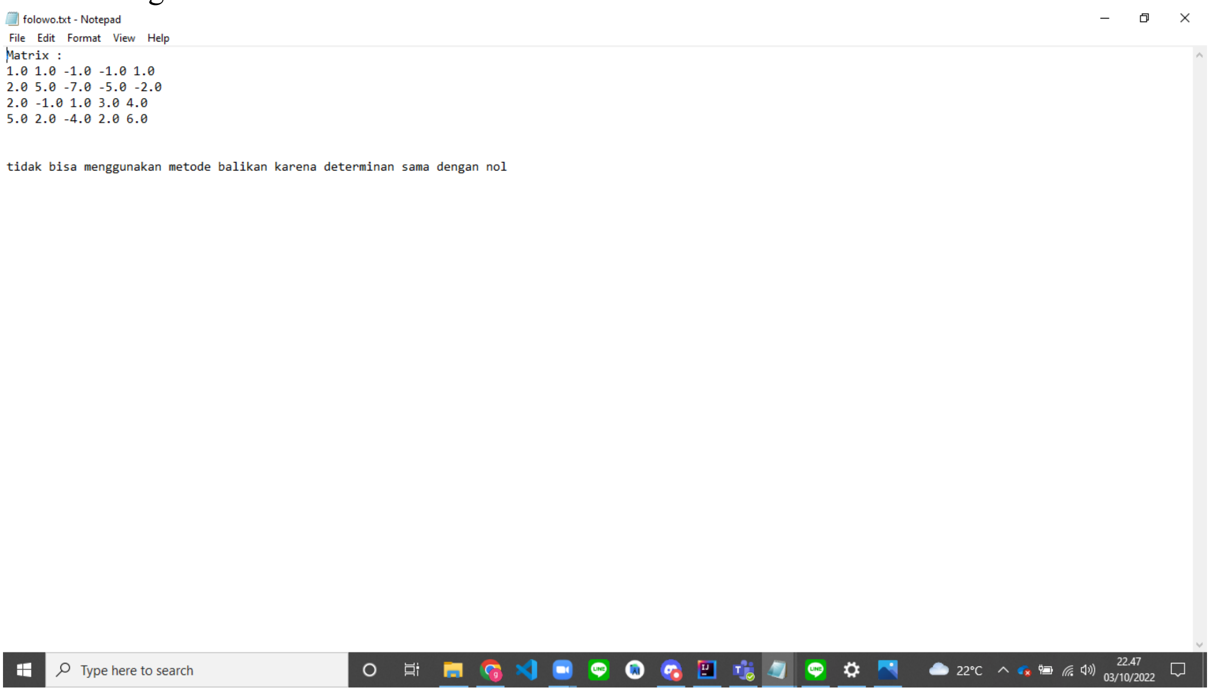
```
file12.txt - Notepad
File  Edit  Format  View  Help
Matrix :
1.0 1.0 -1.0 -1.0 1.0
2.0 5.0 -7.0 -5.0 -2.0
2.0 -1.0 1.0 3.0 4.0
5.0 2.0 -4.0 2.0 6.0


matrix ini tidak memenuhi SPL Gauss
```

Type here to search 22°C 22.42 03/10/2022

Dengan kaidah gauss jordan



Dengan kaidah balikan



4.1.4 Studi Kasus 4

n = 6

Dengan kaidah Cramer

```
C:\Windows\System32\cmd.exe - java  -cp ..\bin main
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 1
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:6
masukkan jumlah kolom matrix:7
masukkan nama file txt:
..\test\soal\StudiKasus1d.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus1dCramer_1.txt


===========================================CRAMER============================================
Matrix :
1.0 0.5 0.3333 0.25 0.2 0.1667 1.0
0.5 0.3333 0.25 0.2 0.1667 0.1429 0.0
0.3333 0.25 0.2 0.1667 0.1429 0.125 0.0
0.25 0.2 0.1667 0.1429 0.125 0.1111 0.0
0.2 0.1667 0.1429 0.125 0.1111 0.1 0.0
0.1667 0.1429 0.125 0.1111 0.1 0.0909 0.0
dengan metode cramer, persamaan ini memilki hasil:
x1 = 9.596501658225085
x2 = -33.54912538246162
x3 = 32.4250948818041
x4 = -210.507601150591
x5 = 522.6861576156678
x6 = -327.17189156054684
============================================================================================


matrix berhasil diproses
```

Dengan kaidah gauss

```
C:\Windows\System32\cmd.exe - java -cp ..\bin main
Pilih menu : 1
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:6
masukkan jumlah kolom matrix:7
masukkan nama file txt:
..\test\soal\StudiKasus1d_1.txt
masukkan nama lokasi penyimpanan beserta nama txt: ..\test\hasil\StudiKasus1d_1Gauss.txt


===========================================GAUSS SPL===========================================
Matrix:
1.0 0.5 0.3333 0.25 0.2 0.1667 1.0
0.5 0.3333 0.25 0.2 0.1667 0.1429 0.0
0.3333 0.25 0.2 0.1667 0.1429 0.125 0.0
0.25 0.2 0.1667 0.1429 0.125 0.1111 0.0
0.2 0.1667 0.1429 0.125 0.1111 0.1 0.0
0.1667 0.1429 0.125 0.1111 0.1 0.0909 0.0
dengan gauss:
1.0 0.5 0.3333 0.25 0.2 0.1667 1.0
0.0 1.0 1.0006002400960388 0.9003601440576233 0.800720288115246 0.7148859543817528 -6.002400960384155
0.0 0.0 1.0 1.5114972039858463 1.7237935225560266 1.7878792766107656 30.302612266887188
0.0 0.0 0.0 1.0 2.0790802456077917 3.2438207977515487 -185.08812279126005
0.0 0.0 0.0 0.0 1.0 0.9255602209413397 219.86886920269043
0.0 0.0 0.0 0.0 0.0 1.0 -327.17189156007066
memiliki hasil:
x1 = 9.596501644377733
x2 = -33.54912541496245
x3 = 32.4250950261486
x4 = -210.50760109685274
x5 = 522.6861574408255
x6 = -327.17189156007066
===============================================================================================
```

Dengan Gauss Jordan

```
C:\Windows\System32\cmd.exe - java -cp ..\bin main
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer
Pilih menu : 2
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:6
masukkan jumlah kolom matrix:7
masukkan nama file txt:
..\test\soal\StudiKasus1d_1.txt
masukkan nama lokasi penyimpanan beserta nama txt: ..\test\hasil\StudiKasus1d_1GaussJordan.txt


===========================================GAUSS JORDAN===========================================
dengan gauss jordan:
1.0 0.0 0.0 0.0 0.0 0.0 9.596501644377733
0.0 1.0 0.0 0.0 0.0 0.0 -33.54912541496245
0.0 0.0 1.0 0.0 0.0 0.0 32.4250950261486
0.0 0.0 0.0 1.0 0.0 0.0 -210.50760109685274
0.0 0.0 0.0 0.0 1.0 0.0 522.6861574408255
0.0 0.0 0.0 0.0 0.0 1.0 -327.17189156007066
memiliki hasil:
x1 = 9.596501644377733
x2 = -33.54912541496245
x3 = 32.4250950261486
x4 = -210.50760109685274
x5 = 522.6861574408255
x6 = -327.17189156007066
===============================================================================================


matrix berhasil diproses
```

Dengan Matriks balikan

```
C:\Windows\System32\cmd.exe - java  -cp ..\bin main
masukkan jumlah baris matrix:6
masukkan jumlah kolom matrix:7
masukkan nama file txt:
..\test\soal\StudiKasus1d_1.txt
masukkan nama lokasi penyimpanan beserta nama txt: ..\test\hasil\StudiKasus1d_1Balikan.txt


===========================================INVERSE SPL===========================================
Matrix:
1.0 0.5 0.3333 0.25 0.2 0.1667 1.0
0.5 0.3333 0.25 0.2 0.1667 0.1429 0.0
0.3333 0.25 0.2 0.1667 0.1429 0.125 0.0
0.25 0.2 0.1667 0.1429 0.125 0.1111 0.0
0.2 0.1667 0.1429 0.125 0.1111 0.1 0.0
0.1667 0.1429 0.125 0.1111 0.1 0.0909 0.0

Memiliki inverse:
9.596501644377733 -33.54912541496259 32.425095026149165 -210.50760109685416 522.6861574408263 -327.1718915600709
-33.54912541496245 90.91295553422515 -363.2680690855959 3678.2153106775113 -7866.36140296172 4576.414805021278
32.4250950261486 -363.26806908559774 4014.640581447351 -17615.028945625927 27747.610686549757 -14005.013000499268
-210.50760109685274 3678.215310677504 -17615.028945625923 35058.722130359914 -31158.46749940108 10254.961371478466
522.6861574408255 -7866.361402961718 27747.61068654977 -31158.46749940108 3763.5635843976097 7193.281704761528
-327.17189156007066 4576.414805021277 -14005.013000499273 10254.961371478461 7193.281704761526 -7771.813807475013

dengan result
x1 = 9.596501644377733
x2 = -33.54912541496245
x3 = 32.4250950261486
x4 = -210.50760109685274
x5 = 522.6861574408255
x6 = -327.17189156007066
=================================================================================================


matrix berhasil diproses
```

n = 10

Dengan kaidah Cramer

```
C:\Windows\System32\cmd.exe - java -cp ..\bin main
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 1
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:10
masukkan jumlah kolom matrix:11
masukkan nama file txt:
..\test\soal\StudiKasus1d_2.txt
masukkan nama lokasi penyimpanan beserta nama txt: ..\test\hasil\StudiKasus1d_2Cramer.txt


===========================================CRAMER===========================================
Matrix :
1.0 0.5 0.33333334 0.25 0.2 0.16666667 0.14285715 0.125 0.11111111 0.1 1.0
0.5 0.33333334 0.25 0.2 0.16666667 0.14285715 0.125 0.11111111 0.1 0.09090909 0.0
0.33333334 0.25 0.2 0.16666667 0.14285715 0.125 0.11111111 0.1 0.09090909 0.083333336 0.0
0.25 0.2 0.16666667 0.14285715 0.125 0.11111111 0.1 0.09090909 0.083333336 0.07692308 0.0
0.2 0.16666667 0.14285715 0.125 0.11111111 0.1 0.09090909 0.083333336 0.07692308 0.071428575 0.0
0.16666667 0.14285715 0.125 0.11111111 0.1 0.09090909 0.083333336 0.07692308 0.071428575 0.06666667 0.0
0.14285715 0.125 0.11111111 0.1 0.09090909 0.083333336 0.07692308 0.071428575 0.06666667 0.0625 0.0
0.125 0.11111111 0.1 0.09090909 0.083333336 0.07692308 0.071428575 0.06666667 0.0625 0.05882353 0.0
0.11111111 0.1 0.09090909 0.083333336 0.07692308 0.071428575 0.06666667 0.0625 0.05882353 0.055555556 0.0
0.1 0.09090909 0.083333336 0.07692308 0.071428575 0.06666667 0.0625 0.05882353 0.055555556 0.05263158 0.0
dengan metode cramer, persamaan ini memilki hasil:
x1 = 2.282913390151596 48E17
x2 = -3.0001551158698609E18
x3 = 8.4484475005611571E18
x4 = 1.95017174245482906E18
x5 = -2.2130781716001317E19
x6 = 5.1110725074691922E18
x7 = 4.8181402591593011E18
x8 = 1.7405587883222856E19
x9 = -8.0657737756649894E17
x10 = -1.2248430945148256E19
```

4.2 SPL bentuk matriks augmented

4.2.1 Studi Kasus 1

Dengan metode gauss

```
==========================================GAUSS SPL==========================================
Matrix:
1.0 -1.0 2.0 -1.0 -1.0
2.0 1.0 -2.0 -2.0 -2.0
-1.0 2.0 -4.0 1.0 1.0
3.0 0.0 0.0 -3.0 -3.0
Exception in thread "main" java.lang.NullPointerException: Cannot store to int array because "<local1>" is null
        at Operator.whichRowNoll(Operator.java:308)
        at TxT.writeGaussSPL(TxT.java:220)
        at main.main(main.java:48)
```

Dengan metode gauss jordan

```
==========================================GAUSS JORDAN==========================================
Exception in thread "main" java.lang.NullPointerException: Cannot store to int array because "<local1>" is null
        at Operator.whichRowNoll(Operator.java:308)
        at TxT.writeGaussJordanSPL(TxT.java:276)
        at main.main(main.java:75)
```

Dengan metode invers

```
===========================================INVERSE SPL===========================================
1.0 -1.0 2.0 -1.0 -1.0
2.0 1.0 -2.0 -2.0 -2.0
-1.0 2.0 -4.0 1.0 1.0
3.0 0.0 0.0 -3.0 -3.0

tidak bisa menggunakan metode balikan karena determinan sama dengan nol
=================================================================================================
```

Dengan metode cramer

```
===========================================CRAMER===========================================
Matrix :
1.0 -1.0 2.0 -1.0 -1.0
2.0 1.0 -2.0 -2.0 -2.0
-1.0 2.0 -4.0 1.0 1.0
3.0 0.0 0.0 -3.0 -3.0
dengan metode cramer, persamaan ini memilki hasil:
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 4 out of bounds for length 4
        at DeterminanOBE.swapRow(DeterminanOBE.java:84)
        at DeterminanOBE.obe(DeterminanOBE.java:98)
        at DeterminanOBE.determinan(DeterminanOBE.java:130)
        at Cramer.cramer(Cramer.java:21)
        at TxT.writeCramer(TxT.java:312)
        at main.main(main.java:129)
```

4.2.2 Studi Kasus 2

Dengan metode gauss

```
masukkan nama file txt:
..\test\soal\StudiKasus2b.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus2bGauss.txt


===========================================GAUSS SPL===========================================
Matrix:
2.0 0.0 8.0 0.0 8.0
0.0 1.0 0.0 4.0 6.0
-4.0 0.0 6.0 0.0 6.0
0.0 -2.0 0.0 3.0 -1.0
2.0 0.0 -4.0 0.0 -4.0
0.0 1.0 0.0 -2.0 0.0
===============================================================================================


matrix gagal diproses
```

Dengan metode gauss jordan

```
===========================================GAUSS JORDAN===========================================
dengan gauss jordan:
1.0 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 1.0 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 1.0 0.0
memiliki hasil:
x1 = NaN
x2 = NaN
x3 = NaN
x4 = NaN
x5 = NaN
=================================================================================================


matrix berhasil diproses
```

Dengan metode matriks balikan

```
===========================================INVERSE SPL===========================================
Matrix:
2.0 0.0 8.0 0.0 8.0 0.0
1.0 0.0 4.0 6.0 -4.0 0.0
6.0 0.0 6.0 0.0 -2.0 0.0
3.0 -1.0 2.0 0.0 -4.0 0.0
-4.0 0.0 1.0 0.0 -2.0 0.0

Memiliki inverse:
-0.02631578947368418 0.0 0.06315789473684211 0.0 -0.16842105263157894
-0.28947368421052616 0.0 0.6947368421052631 -1.0 0.14736842105263157
0.05263157894736842 0.0 0.0736842105263158 0.0 0.1368421052631579
0.021929824561403508 0.16666666666666666 -0.11929824561403508 0.0 -0.12631578947368421
0.07894736842105263 0.0 -0.08947368421052632 0.0 -0.09473684210526316

dengan result
x1 = 0.0
x2 = 0.0
x3 = 0.0
x4 = 0.0
x5 = 0.0
=================================================================================================
```

Dengan kaidah cramer

```
============================================CRAMER============================================
Matrix :
2.0 0.0 8.0 0.0 8.0 0.0
1.0 0.0 4.0 6.0 -4.0 0.0
6.0 0.0 6.0 0.0 -2.0 0.0
3.0 -1.0 2.0 0.0 -4.0 0.0
-4.0 0.0 1.0 0.0 -2.0 0.0
dengan metode cramer, persamaan ini memilki hasil:
x1 = 0.0
x2 = 0.0
x3 = 0.0
x4 = 0.0
x5 = 0.0
==============================================================================================


matrix berhasil diproses
```

4.3 SPL bentuk persamaan

4.3.1 Studi Kasus 1

Dengan kaidah Cramer

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 1
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:6
masukkan jumlah kolom matrix:5
matrix tidak valid untuk dijadikan gauss, masukkan kembali data yang valid
masukkan jumlah baris matrix:4
masukkan jumlah kolom matrix:5
masukkan nama file txt:
..\test\Soal\StudiKasus3a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/Hasil/StudiKasus3aCramer.txt


===========================================CRAMER===========================================
Matrix :
8.0 1.0 3.0 2.0 0.0
2.0 9.0 -1.0 -1.0 1.0
1.0 3.0 2.0 3.0 -1.0
2.0 1.0 0.0 3.0 6.0
dengan metode cramer, persamaan ini memilki hasil:
x1 = 0.7301293900184843
x2 = -0.20147874306839186
x3 = -2.933456561922366
x4 = 1.5804066543438078
============================================================================================


matrix berhasil diproses
MENU
```

Dengan matriks balikan

```
==========================================INVERSE SPL==========================================
Matrix:
8.0 1.0 3.0 2.0 0.0
2.0 9.0 -1.0 -1.0 1.0
1.0 3.0 2.0 3.0 -1.0
2.0 1.0 0.0 3.0 6.0

Memiliki inverse:
0.11460258780036968 0.02957486136783734 -0.15711645101663588 0.09057301293900184
-0.024029574861367836 0.09057301293900184 0.08133086876155268 -0.035120147874306683
0.08133086876155268 -0.07578558225508317 0.34011090573012936 -0.4195933456561922
-0.06839186691312385 -0.04990757855822551 0.07763401109057301 0.2846580406654344

dengan result
x1 = 0.7301293900184843
x2 = -0.2014787430683918
x3 = -2.933456561922366
x4 = 1.5804066543438076
===============================================================================================


matrix berhasil diproses
```

Dengan metode gauss jordan

```
..\test\soal\StudiKasus3a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus3aGaussJordan.txt


==========================================GAUSS JORDAN==========================================
dengan gauss jordan:
1.0 0.0 0.0 0.0 0.7301293900184843
0.0 1.0 0.0 0.0 -0.20147874306839186
0.0 0.0 1.0 0.0 -2.933456561922366
0.0 0.0 0.0 1.0 1.5804066543438078
memiliki hasil:
x1 = 0.7301293900184843
x2 = -0.20147874306839186
x3 = -2.933456561922366
x4 = 1.5804066543438078
===============================================================================================


matrix berhasil diproses
```

Dengan metode gauss

```
==========================================GAUSS SPL==========================================
Matrix:
8.0 1.0 3.0 2.0 0.0
2.0 9.0 -1.0 -1.0 1.0
1.0 3.0 2.0 3.0 -1.0
2.0 1.0 0.0 3.0 6.0
dengan gauss:
1.0 0.125 0.375 0.25 0.0
0.0 1.0 -0.2 -0.17142857142857143 0.11428571428571428
0.0 0.0 1.0 1.474025974025974 -0.6038961038961039
0.0 0.0 0.0 1.0 1.5804066543438078
memiliki hasil:
x1 = 0.7301293900184843
x2 = -0.20147874306839186
x3 = -2.933456561922366
x4 = 1.5804066543438078
===============================================================================================


matrix berhasil diproses
```

4.3.2 Studi Kasus 2

Dengan metode gauss

```
==========================================GAUSS SPL==========================================
Matrix:
0.0 0.0 0.0 0.0 0.0 0.0 1.0 1.0 1.0 13.0
0.0 0.0 0.0 1.0 1.0 1.0 0.0 0.0 0.0 15.0
1.0 1.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 8.0
0.0 0.0 0.04289 0.0 0.04289 0.75 0.04289 0.75 0.61396 14.79
0.0 0.25 0.91421 0.25 0.91421 0.25 0.91421 0.25 0.0 14.31
0.61396 0.75 0.04289 0.75 0.04289 0.0 0.04289 0.0 0.0 3.81
0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 18.0
0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 12.0
1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 6.0
0.04289 0.75 0.61396 0.0 0.04289 0.75 0.0 0.0 0.04289 10.51
0.91421 0.25 0.0 0.25 0.91421 0.25 0.0 0.25 0.91421 16.13
=============================================================================================


matrix gagal diproses
```

Dengan metode gauss jordan

```
==========================================GAUSS JORDAN==========================================
=============================================================================================


matrix gagal diproses
```

Dengan matriks balikan

```
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus3bInvers.txt


==========================================INVERSE SPL==========================================
matrix yang dimasukkan tidak nx(n+1)
=============================================================================================


matrix gagal diproses
```

Dengan kaidah cramer

```
==========================================CRAMER==========================================
Matrix :
0.0 0.0 0.0 0.0 0.0 0.0 1.0 1.0 1.0 13.0
0.0 0.0 0.0 1.0 1.0 1.0 0.0 0.0 0.0 15.0
1.0 1.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 8.0
0.0 0.0 0.04289 0.0 0.04289 0.75 0.04289 0.75 0.61396 14.79
0.0 0.25 0.91421 0.25 0.91421 0.25 0.91421 0.25 0.0 14.31
0.61396 0.75 0.04289 0.75 0.04289 0.0 0.04289 0.0 0.0 3.81
0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 18.0
0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 12.0
1.0 0.0 0.0 1.0 0.0 0.0 1.0 0.0 0.0 6.0
0.04289 0.75 0.61396 0.0 0.04289 0.75 0.0 0.0 0.04289 10.51
0.91421 0.25 0.0 0.25 0.91421 0.25 0.0 0.25 0.91421 16.13
dengan metode cramer, persamaan ini memilki hasil:
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 9 out of bounds for length 9
        at DeterminanOBE.All0rowcol(DeterminanOBE.java:24)
        at DeterminanOBE.HasAll0RowCol(DeterminanOBE.java:10)
        at DeterminanOBE.determinan(DeterminanOBE.java:123)
        at Cramer.cramer(Cramer.java:21)
        at TxT.writeCramer(TxT.java:312)
        at main.main(main.java:129)
```

4.4 Interpolasi Polinom

4.4.1 Studi kasus 1

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:7
masukkan nama file txt:
..\test\testcase_4a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4aR1.txt
Matrix:
0.4 0.043
0.7 0.005
0.11 0.058
0.14 0.072
0.17 0.1
0.2 0.13
0.23 0.147
x = 0.2

memiliki hasil interpolasi polinom: 0.12999999999985423
matrix berhasil diproses
```

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:7
masukkan nama file txt:
..\test\testcase_4a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4aR2.txt
Matrix:
0.4 0.043
0.7 0.005
0.11 0.058
0.14 0.072
0.17 0.1
0.2 0.13
0.23 0.147
x = 0.55

memiliki hasil interpolasi polinom: 2.1375716211232714
matrix berhasil diproses
```

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:7
masukkan nama file txt:
..\test\testcase_4a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4aR3.txt
Matrix:
0.4 0.043
0.7 0.005
0.11 0.058
0.14 0.072
0.17 0.1
0.2 0.13
0.23 0.147
x = 0.85

memiliki hasil interpolasi polinom: -66.26963932203739
matrix berhasil diproses
```

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:7
masukkan nama file txt:
..\test\testcase_4a.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4aR2.txt
Matrix:
0.4 0.043
0.7 0.005
0.11 0.058
0.14 0.072
0.17 0.1
0.2 0.13
0.23 0.147
x = 1.28

memiliki hasil interpolasi polinom: -3485.144901976004
matrix berhasil diproses
```

Hasil Analisis:

Pada studi kasus 4.4.1 program menerima input sebuah file atau input yang berisi sebuah matriks yang menandakan x dan y. Kemudian dengan menggunakan interpolasi didapatkan sebuah fungsi polinomial untuk menghitung beberapa nilai yaitu :

f(0.2) = 0.1299
f(0.55) = 2.1375
f(0.85) = -66.269
f(1.28) = -3485.1229

4.4.2 Studi Kasus 2

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:10
masukkan nama file txt:
..\test\testcase_4b.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4aR2.txt
Matrix:
6.567 12624.0
7.0 21807.0
7.258 38391.0
7.451 54517.0
7.548 51952.0
7.839 28228.0
8.161 35764.0
8.484 20813.0
8.709 12408.0
9.0 10534.0
x = 7.516

memiliki hasil interpolasi polinom: 53540.654296875
matrix berhasil diproses
```

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:10
masukkan nama file txt:
..\test\testcase_4b.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4bR2.txt
Matrix:
6.567 12624.0
7.0 21807.0
7.258 38391.0
7.451 54517.0
7.548 51952.0
7.839 28228.0
8.161 35764.0
8.484 20813.0
8.709 12408.0
9.0 10534.0
x = 8.326

memiliki hasil interpolasi polinom: 36146.76953125
matrix berhasil diproses
```

```
MENU
1. Sistem Persamaan Linier
2. Determinan
3. Matriks balikan
4. Interpolasi Polinom
5. Interpolasi Bicubic
6. Regresi linier berganda
7. Keluar
Pilih menu : 4
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan jumlah baris matrix:10
masukkan nama file txt:
..\test\testcase_4b.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/res/testcase4bR3.txt
Matrix:
6.567 12624.0
7.0 21807.0
7.258 38391.0
7.451 54517.0
7.548 51952.0
7.839 28228.0
8.161 35764.0
8.484 20813.0
8.709 12408.0
9.0 10534.0
x = 9.167

memiliki hasil interpolasi polinom: -667734.4375
matrix berhasil diproses
```

Hasil Analisis :
Pada hasil studi kasus 4.4.1 didapatkan hasil dari tanggal - tanggal setelah diubah menjadi desimal sebagai berikut:
f(7.516) = 5354.65
f(8.326) = 36146.77

f(9.167) = -667734.4375
Hasil dari interpolasi berikut juga dipengaruhi oleh overflow dari operasi.

4.3 Studi Kasus 3

```
=========================================INTERPOLASI POLINOM=========================================
Matrix:
0.0 0.0
0.1 0.2706900417
0.2 0.3427695582
0.3 0.3865315957
0.4 0.4188842301
0.5 0.4454308682
0.6 0.4684314696
0.7 0.4888654824
0.8 0.5071579685
0.9 0.5234794825
1.0 0.5378828427
1.1 0.5503697548
1.2 0.5609246748
1.3 0.5695323935
1.4 0.5761871198
1.5 0.5808969391
1.6 0.5836856613
1.7 0.5845931468
1.8 0.5836747201
1.9 0.5810000238
f(x) = 0.0+5.496498065513451x^1-44.99492622573025x^2+231.6714586091847x^3-735.2360024398392x^4+1444.1477680150583x^5-170
8.0868514948434x^6+1113.1515945450867x^7-306.69473536753486x^8+0.0x^9+0.0x^10+0.0x^11+0.0x^12+0.0x^13+0.0x^14+0.0x^15+0.
0x^16+0.0x^17+0.0x^18+0.0x^19
x = 1

memiliki hasil interpolasi polinom: -0.5451962931045387
=====================================================================================================
```

Dari hasil penyederhanaan fungsi dengan interpolasi didapatkan fungsi

$$f(x) = 5.5x - 45x^2 + 231.67x^3 - 735.24x^4 + 1444.15x^5 - 1708.08x^6 + 1113.15x^7 - 306.694x^8$$

4.5 Interpolasi Bicubic

4.5.1 Studi Kasus 1

```
Pilih menu : 5
pilih pembaca matrix
1. baca melalui keyboard
2. baca melalui file txt
masukkan pilihan: 2
masukkan nama file txt:
..\test\soal\StudiKasus4.1.txt
masukkan nama lokasi penyimpanan beserta nama txt: ..\test\soal\StudiKasus4.1.txt


=========================================INTERPOLASI BIKUBIK==========================
Matrix:
153.0 59.0 210.0 96.0
125.0 161.0 72.0 81.0
98.0 101.0 42.0 12.0
21.0 51.0 0.0 16.0


memiliki hasil interpolasi bikubik: 161.0
=====================================================================================


matrix berhasil diproses
```

```
masukkan nama file txt:
..\test\soal\StudiKasus4.2.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus4.2.txt


===========================================INTERPOLASI BIKUBIK===========================================
Matrix:
153.0 59.0 210.0 96.0
125.0 161.0 72.0 81.0
98.0 101.0 42.0 12.0
21.0 51.0 0.0 16.0


memiliki hasil interpolasi bikubik: 97.72656249999997
=========================================================================================================
```

```
masukkan pilihan: 2
masukkan nama file txt:
..\test\soal\StudiKasus4.3.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus4.3.txt


===========================================INTERPOLASI BIKUBIK===========================================
Matrix:
153.0 59.0 210.0 96.0
125.0 161.0 72.0 81.0
98.0 101.0 42.0 12.0
21.0 51.0 0.0 16.0


memiliki hasil interpolasi bikubik: 82.5020751953125
=========================================================================================================
```

```
masukkan nama file txt:
..\test\soal\StudiKasus4.4.txt
masukkan nama lokasi penyimpanan beserta nama txt: ../test/hasil/StudiKasus4.4.txt


===========================================INTERPOLASI BIKUBIK===========================================
Matrix:
153.0 59.0 210.0 96.0
125.0 161.0 72.0 81.0
98.0 101.0 42.0 12.0
21.0 51.0 0.0 16.0


memiliki hasil interpolasi bikubik: 48.09649999999998
=========================================================================================================
```

Dari matriks 4x4 yang diberikan, didapatkan beberapa interpolasi yang berada di antara 1-0 yaitu :

f(0,0) = 161

f(0.5,0.5) = 97.73

f(0.25,0.75) = 82.50

f(0.1,0.9) = 48.096

4.6 Regresi Linier Berganda

4.5.1 Studi Kasus

```
C:\Windows\System32\cmd.exe - java  -cp ..\bin main
===========================================REGRESI===========================================
Matrix:
72.4 76.3 29.18 0.9
41.6 76.3 29.35 0.91
34.3 70.3 29.24 0.96
35.1 77.1 29.27 0.89
10.7 68.0 29.78 1.0
12.9 67.4 29.39 1.1
8.3 66.8 29.69 1.1
20.1 76.9 29.48 1.15
72.2 77.7 29.09 1.03
24.0 67.7 29.6 0.77
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 86.0 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
memiliki regresi:
20.0 863.0999999999999 1527.7000000000003 587.8399999999999 19.45
863.0999999999999 54876.89 67218.16 25283.395 793.0459999999999
1527.7000000000003 67218.16 117493.01000000001 44892.912 1485.2479999999998
587.8399999999999 25283.395 44892.912 17278.508600000005 571.8584000000001
=============================================================================================
```

BAB V
KESIMPULAN, SARAN, DAN REFLEKSI

5.1 Kesimpulan

Dari program java yang telah dibuat didapatkan beberapa kesimpulan:

- Melalui teori dalam aljabar linier didapatkan sebuah aplikasi yang mampu menerima input melalui keyboard dan file .txt.
- Program mampu menghitung solusi dari Sistem Persamaan Linier dengan menggunakan beberapa metode yaitu eliminasi Gauss, eliminasi Gauss Jordan, metode matriks balikan, dan metode Cramer.
- Program dapat menghitung determinan dengan reduksi baris dan juga ekspansi kofaktor.
- Program dapat menghitung matriks balikan dengan metode obe dan adjoin.
- Program mampu menyelesaikan permasalahan interpolasi polinom, bicubic dan regresi linier.

5.2 Saran

Tugas besar ini menambah semangat untuk bereksplorasi secara mandiri, kedepannya apabila diadakan tugas besar dengan waktu yang lebih banyak tentunya waktu untuk eksplorasi akan menjadi lebih banyak pula. Dikarenakan perhitungan - perhitungan yang terlibat dalam setiap operasi harus selalu diperhatikan agar presisi sehingga tidak mempengaruhi perhitungan yang lain juga dikarenakan fungsi - fungsi dasar seperti gauss, gauss jordan, determinan sangat sering untuk dipanggil ke dalam fungsi lain.

5.3 Refleksi

Dari tugas besar ini kami mendapati beberapa kesalahan seperti kurang familiernya dengan syntax dari program java, kurangnya pemahaman mengenai beberapa teori seperti interpolasi bicubic, kesalahan dalam memilih tipe data sehingga mengakibatkan error code. Kesalahan - kesalahan tersebut murni sebagai kesalahan kami dan dari kesalahan tersebut kami juga mendapat ilmu dan pengalaman baru di dalam memecahkan suatu solusi.

REFERENSI

Urrutia, J. D., Olfindo, M. T., Mercado, J., Baygan, A. V., & Baccay, E. (n.d.). European Academic Research. *A Mathematical Model for Estimating Imports and Exports in the Philippines: A Normal Estimation Equation for Multiple Linear Regression*, *3*(2), 12.

https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/algeo.htm

https://www.ece.mcmaster.ca/~xwu/interp_1.pdf