Tugas Besar 1 IF2123 Aljabar Linear dan Geometri

Aplikasi Matriks dalam Sistem Persamaan Linear, Determinan, dan Aplikasinya

Maria Flora Renata Siringoringo (13522010)
M Athaullah Daffa Kusuma M (13522044)
Dzaky Satrio Nugroho (13522059)

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2023

Daftar Isi

Bab 1: Deskripsi Masalah

Sistem persamaan linier (SPL) banyak ditemukan di dalam bidang sains dan rekayasa. Penulis 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 ini, penulis 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, library tersebut digunakan di dalam program Java untuk menyelesaikan berbagai persoalan yang dimodelkan dalam bentuk SPL, menyelesaikan persoalan interpolasi, dan persoalan regresi.

Spesifikasi programnya adalah sebagai berikut:

1. 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. Seperti contoh di bawah ini.

```
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 aij. Masukan dari file berbentuk matriks, setiap elemen matriks dipisah oleh spasi. Misalnya,

```
3 4.5 2.8
-3 7 8.3 1
0.5 -10 -9
```

luaran (output) dari determinan dan matriks balikan disesuaikan dengan persoalan masing-masing.

3. 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
```

4. 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.

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 x4 = -2, x3 = 2s – t, x2 = s, dan x1 = 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

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

dan untuk regresi adalah

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

7. Untuk persoalan bicubic spline interpolation, masukan dari file text (.txt) yang berisi matriks berukuran 4 x 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), fx(0, 0), fx(1, 0), fx(0, 1), fx(1, 1), fy(0, 0), fy(1, 0), fy(0, 1), fy(1, 1), fxy(0, 0), fxy(1, 0), fxy(0, 1), fxy(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 Persamaaan 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

2.1. Sistem Persamaan Linear

Sistem Persamaan Linear adalah

2.1.1. Metode Eliminasi Gauss

Metode eliminasi Gauss merupakan metode yang digunakan untuk mengeliminasi sistem persamaan linear yang diterapkan menggunakan matriks *augmented*. Metode ini memanfaatkan operasi baris elementer (OBE) dengan ketentuan:

- Mengalikan sebuah baris dengan konstanta tidak nol
- Menukar dua buah baris
- Menambah sebuah baris dengan kelipatan baris lainnya

Metode eliminasi Gauss dilakukan hingga matriks *augmented* menjadi sebuah matriks eselon (matriks yang memiliki 1 utama pada setiap baris, kecuali baris yang seluruhnya nol, jika ada baris yang seluruhnya nol, maka semua baris itu dikumpulkan pada bagian bawah matriks.). Setelah menjadi matriks eselon, akan diambil hasil akhir dari setiap variabel yang ada dengan melakukan substitusi mundur. Bentuk matriks eselon baris adalah sebagai berikut.

$$\begin{matrix} 1 & \dots & \dots & \dots \\ 0 & 1 & \dots & \dots \\ 0 & 0 & 1 & \dots \\ 0 & 0 & 0 & 1 \end{matrix}$$

Ada tiga kemungkinan solusi metode eliminasi Gauss:

1. Solusi unik (hanya satu solusi), yaitu ketika baris terakhir tidak 0 semua,
2. Solusi banyak/tidak berhingga, ketika baris terakhir 0 semua sehingga hasil yang didapat akan berbentuk parametrik,
3. Tidak ada solusi, ketika ada persamaan dimana semua konstanta variabel bernilai 0, namun solusinya bukan 0.

2.1.2. Metode Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan merupakan metode yang digunakan untuk mengeliminasi sistem persamaan linear yang diterapkan menggunakan matriks *augmented*. Sama halnya seperti Metode Eliminasi Gauss, Metode Eliminasi Gauss-Jordan juga memanfaatkan operasi baris elementer (OBE), akan tetapi tujuan akhirnya adalah membentuk Matriks eselon tereduksi (matriks yang memiliki 1 utama pada setiap baris, dan angka di atas dan di bawah satu utama merupakan angka nol). Bentuk Matriks eselon tereduksi adalah sebagai berikut.

$$\begin{matrix} 1 & 0 & 0 & 0 & \dots \\ 0 & 1 & 0 & 0 & \dots \\ 0 & 0 & 1 & 0 & \dots \\ 0 & 0 & 0 & 1 & \dots \end{matrix}$$

Selanjutnya, akan dilakukan proses seperti Metode Eliminasi Gauss.

2.1.3. Metode Matriks Balikan

Metode Matriks Balikan juga salah satu cara untuk menyelesaikan sebuah SPL. Tinjau SPL Ax = b, kemudian kalikan kedua ruas persamaan dengan A^{-1}.

$$(A^{-1})Ax = (A^{-1})b$$
$$Ix = A^{-1}b$$
$$x = A^{-1}b$$

Jadi solusi SPL Ax = b adalah $x = A^{-1}b$.

2.1.4. Metode Cramer

Jika Ax = b adalah SPL dengan n peubah dan n persamaan dan determinan A bukan 0, maka SPL tersebut memiliki solusi yang unik yaitu:

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

Yang dalam hal ini, A_i adalah matriks yang diperoleh dengan cara mengganti kolom ke-i pada matriks A dengan b.

2.2. Determinan

Determinan merupakan sebuah nilai yang dapat dihitung dari elemen-elemen sebuah matriks persegi. Notasi untuk memanggil determinan dari sebuah matriks adalah seperti det(A), atau | A |. Untuk berbagai ukuran matriks, terdapat cara yang berbeda dalam menghitung determinannya, seperti untuk Matriks 1 x 1, determinannya merupakan elemen dari matriks itu sendiri. Untuk ukuran matriks lainnya, banyak metoda yang dapat digunakan untuk menemukan determinan dari matriks tersebut, yang akan dibahas di sini yaitu metode reduksi baris dan metode ekspansi kofaktor.

2.2.1. Metode Reduksi Baris

Metode reduksi baris yaitu metode yang membentuk sebuah matriks persegi sembarang menjadi sebuah matriks segitiga atas ataupun segitiga bawah. Cara mengubahnya yaitu dengan metode operasi baris elementer, tetapi tidak diperlukan untuk mengalikan baris dengan sebuah konstanta, cukup menambah/mengurangi baris dengan kelipatan baris lainnya, dan menukar posisi dari 2 baris. Setelah menjadi matriks segitiga atas atau matriks segitiga bawah, determinan didapat dengan rumus sebagai berikut.

$$det(A) = -1^{p} \times a_{11} \times a_{22} \times ... \times a_{nn}$$

Dengan p = banyaknya pertukaran baris yang dilakukan

2.2.2 Metode Ekspansi Kofaktor

Metode ekspansi kofaktor dilakukan dengan menghitung minor entri yang didapat dari determinan upa-matriks yang elemen-elemennya tidak berada pada baris i dan kolom j, dan menghitung kofaktor entri dengan mengalikan -1 dipangkatkan i+j dengan minor entri ij. Misalkan matriks A adalah sebagai berikut.

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

Menghitung minor entri baris 1 dan kolom 1 adalah

$$M_{11} = \begin{vmatrix} 2 & -4 \\ 5 & 3 \end{vmatrix} = (2)(3) - (-4)(5) = 26$$

Dan rumus lengkapnya adalah sebagai berikut

$$Det(A) = \sum_{i=1}^{n} a_{ik} M_{ik} \quad atau \quad Det(A) = \sum_{i=1}^{n} a_{ki} M_{ki}$$

Dengan k adalah bilangan bulat sembarang antara i sampai n.

2.3. Matriks Balikan

Sebuah matriks A dikatakan memiliki balikan/invers jika ada sebuah matriks dengan ukuran yang sama dan perkalian kedua matriks tersebut adalah matriks Identitas. Invers dari matriks A ditulis menjadi A^{-1}.

2.3.1. Metode Reduksi Baris

Metode Eliminasi Gauss Jordan dapat digunakan untuk mencari invers dari sebuah matrix. Untuk matriks A ukuran N x N, dapat dicari inversnya dengan cara berikut:

$$[A/I] \; Gauss\; Jordan \Rightarrow \; [I/A^{-1}]$$

Dengan I adalah matriks identitas berukuran N x N.

2.3.2. Metode Ekspansi Kofaktor

Misalkan A adalah matriks N x N dan C_{ij} adalah kofaktor entri a_{ij}. Maka matriks kofaktor dari A adalah

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

Adjoin dari A adalah transpose matriks kofaktor dari A
Adj(A) = transpose matriks kofaktor A

Invers dari matriks A dapat dicari dengan rumus:

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

2.4. Interpolasi Polinom

Interpolasi Polinom adalah prosedur mencari sebuah polinomial berderajat n yang melewati (menginterpolasi) n+1 buah titik yang berbeda. Setelah polinom interpolasi tersebut (misalkan $\mathrm{p}(\mathrm{x})$) didapatkan, $\mathrm{p}(\mathrm{x})$ dapat digunakan untuk menaksir nilai $\mathrm{f}(\mathrm{x})$ yang berada di antara x_0-x_n. Interpolasi polinomial ini memanfaatkan Teori Fundamental Algebra yang berbunyi bahwa semua polinom berderajat n memiliki n buah akar, sehingga derajat perbedaan antara dua polinomial yang melewati n+1 buah titik yang sama, adalah 0. Dari fakta tersebut, didapatkan bahwa untuk setiap kumpulan titik berjumlah n+1, ada sebuah polinom unik berderajat n yang menginterpolasi semua titik tersebut.

Polinom interpolasi derajat n yang menginterpolasi titik-titik $(\mathrm{x}_0, \mathrm{y}_0), (\mathrm{x}_1, \mathrm{y}_1), \ldots, (\mathrm{x}_n, \mathrm{y}_n)$ berbentuk $\mathrm{p}_n(\mathrm{x}) = \mathrm{a}_0 + \mathrm{a}_1\mathrm{x} + \mathrm{a}_2\mathrm{x}^2 + \ldots + \mathrm{a}_n\mathrm{x}^n$. Jika n=2, polinom akan berbentuk linear, jika n=3, maka polinom akan berbentuk kurva, dan seterusnya.

Untuk mencari persamaan polinom tersebut, nilai $(\mathrm{x}_0,\mathrm{y}_0)$ sampai $(\mathrm{x}_n,\mathrm{y}_n)$ disulihkan ke persamaan $\mathrm{p}_n(\mathrm{x}) = \mathrm{a}_0 + \mathrm{a}_1\mathrm{x} + \mathrm{a}_2\mathrm{x}^2 + \ldots + \mathrm{a}_n\mathrm{x}^n$. Akan didapatkan n+1 SPL dengan n+1 variabel yang tidak diketahui (a_0..a_n).

$$\begin{array}{c} a_0 + a_1x_0 + a_2{x_0}^2 + \ldots + a_n\,{x_0}^n = y_0 \\ a_0 + a_1x_1 + a_2{x_1}^2 + \ldots + a_n\,{x_1}^n = y_1 \\ \ldots \qquad\qquad \ldots \\ a_0 + a_1x_n + a_2{x_n}^2 + \ldots + a_n\,{x_n}^n = y_n \end{array}$$

SPL tersebut dapat diselesaikan dengan menggunakan metode Gauss atau metode-metode lain. Setelah mendapatkan nilai a, persamaan $\mathrm{p}_n(\mathrm{x}) = \mathrm{a}_0 + \mathrm{a}_1\mathrm{x} + \mathrm{a}_2\mathrm{x}^2 + \ldots + \mathrm{a}_n\mathrm{x}^n$ dapat dilengkapi dan nilai fungsi pada titik tertentu dapat diestimasi.

2.5. Interpolasi Bicubic Spline

Interpolasi Bicubic Spline adalah metode interpolasi yang menggunakan konsep *spline*, sehingga hasil interpolasi lebih halus atau kurvaturnya lebih kecil. Bicubic Spline Interpolation menggunakan empat titik dan turunan berarah di setiap titik tersebut untuk membentuk polinomial kubik di setiap sel segiempat dari data yang diberikan. Interpolasi ini digunakan untuk perluasan data secqara visual karena lebih akurat dari metode interpolasi lain. Persamaan-persamaan yang digunakan adalah sebagai berikut.

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

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

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

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

Dengan menyulihkan (0,0), (1,0), (0,1) dan (1,1) ke keempat persamaan tersebut, didapatkan sebuah matriks X, yaitu:

$$y = Xa$$

$$\begin{bmatrix} f(0,0) \\ f(1,0) \\ f(0,1) \\ f(1,1) \\ f_x(0,0) \\ f_x(1,0) \\ f_x(0,1) \\ f_x(1,1) \\ f_y(0,0) \\ f_y(1,0) \\ f_y(0,1) \\ f_y(1,1) \\ f_{xy}(0,0) \\ f_{xy}(1,0) \\ f_{xy}(0,1) \\ f_{xy}(1,1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 2 & 4 & 6 & 0 & 3 & 6 & 9 \end{bmatrix} \begin{bmatrix} a_{00} \\ a_{10} \\ a_{20} \\ a_{30} \\ a_{01} \\ a_{11} \\ a_{21} \\ a_{31} \\ a_{02} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{03} \\ a_{13} \\ a_{23} \\ a_{33} \end{bmatrix}$$

Nilai a didapatkan dari men-invers matriks X dan mengalikannya dengan y. Setelah itu, persamaan $f(x, y) = \sum_{j=0}^{3} \sum_{i=0}^{3} a_{ij} x^{i} y^{j}$ dilengkapi untuk mengestimasi nilai fungsi untuk x, di mana x bernilai antara 0-1.

2.6. Regresi Linear Berganda

Regresi Linear Berganda adalah metode yang digunakan untuk memperoleh nilai aproksimasi dari titik tertentu. Metode ini mirip dengan metode interpolasi polinom, bedanya yaitu interpolasi polinom menghasilkan persamaan polinomial, sementara regresi linear berganda menghasilkan persamaan linear dengan n peubah. Rumus umum dari regresi linear berganda adalah sebagai berikut :

$$f(x) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \dots + \beta_k x_{ki} + \epsilon_i$$

Dapat digunakan *Normal Estimation Equation for Multiple Linear Regression* sebagai berikut :

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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

Bab 3: Implementasi pustaka dan program dalam Java

3.1. Matriks (Matrix.java)

File Matriks berisi assignment matriks apabila dipanggil. File Matriks ini digunakan untuk mengolah matriks yang nantinya akan digunakan di file atau fungsi-fungsi lainnya. File Matriks ini juga mengandung method-method yang nantinya digunakan untuk menginput dan mengolah matriks itu sendiri, seperti readMatrix untuk membaca matriks dari keyboard, displayMatrix untuk menampilkan matriks ke terminal output, sampai transposeMatrix untuk mereturn matriks tranpose dan perkalianMatrix untuk mengembalikan hasil kali 2 matriks dari parameter.

3.2. Program Utama (Main.java)

File Main ini sendiri sesuai dengan namanya, yaitu main atau utama. Main berfungsi untuk menampilkan tampilan menu sesuai dengan format. Adapun File ini juga berfungsi untuk mendeclare Scanner di public static void main, selanjutnya Scanner ini akan dioper ke fungsi-fungsi dari file lainnya guna menghindari error. Dari menu yang ditampilkan, user dapat menginput sesuai dengan instruksi sehingga user akan beralih ke menu dari fungsi lainnya. Untuk keluar dari program, user cukup memilih opsi “keluar”.

3.3. Baca Tulis File .txt (TXTReaderWriter.java)

Mekanisme untuk membaca dan menulis file .txt diambil dari Class bawaan java yaitu java.io.File, java.io.FileWriter, dan java.io.FlieNotFoundException. Pembacaan file txt dilakukan dengan fungsi readTXT dengan parameter scanner dan akan mereturn matriks. Pembacaan diawali dengan meminta input dari user berupa nama file txt nya, lalu program akan mencoba untuk membuat tipe data File txt dengan mengambil file yang dipilih dalam format File. Setelah itu, program membaca isi file txt, memangkas spasi dan enter yang ada di file tersebut, lalu memasukkan ke matriks. Sementara untuk penulisan file txt dilakukan dengan fungsi writeTXT yang memiliki parameter Scanner dan String yang akan ditulis di txt yang dipilih.

3.4. Sistem Persamaan Linear

3.4.1. Metode Eliminasi Gauss (Gauss.java)

File ini berisi fungsi f yang menerima input matriks m dan mengembalikan sebuah array of float yang akan digunakan di fungsi lainnya. Program akan mengubah matriks menjadi bentuk matriks eselon, jika ditemukan baris dimana nilai semua peubah 0 dan hasil bukan 0, maupun jumlah baris bukan 0 lebih sedikit dari jumlah peubah, maka program akan mengembalikan array failret. Sebaliknya, akan dilakukan substitusi nilai peubah dari bawah ke atas. Dalam prosesnya juga akan menampilkan hasil SPL di layar dan memberi pilihan apakah user ingin menyimpan jawaban dalam file .txt.

3.4.2. Metode Eliminasi Gauss-Jordan (GaussJordan.java)

File ini berisi fungsi f yang menerima input matriks m dan mengembalikan sebuah array of float. Program akan mengubah matriks menjadi bentuk matriks eselon tereduksi, jika ditemukan baris dimana nilai semua peubah 0 dan hasil bukan 0, maupun jumlah baris bukan 0 lebih sedikit dari jumlah peubah, maka program akan mengembalikan array failret. Sebaliknya, akan dilakukan substitusi nilai peubah dari bawah ke atas. Dalam prosesnya juga akan menampilkan hasil SPL di layar dan memberi pilihan apakah user ingin menyimpan jawaban dalam file .txt.

3.4.3. Metode Matriks Balikan (Balikan.java)

File ini berisi fungsi f yang menerima input matriks m berukuran n x (n+1). Kemudian program akan membagi matriks menjadi 2 buah matriks A berukuran n x n dan matriks B berukuran n x 1. Kemudian matriks A di invers, jika tidak memiliki invers akan muncul pesan error. Matriks A yang sudah di invers kemudian dikalikan dengan matriks B yang menghasilkan matriks baru berukuran 1 x n yang merupakan solusi SPL. Dalam prosesnya juga akan menampilkan hasil SPL di layar dan memberi pilihan apakah user ingin menyimpan jawaban dalam file .txt.

3.4.4. Metode Cramer (Cramer.java)

File ini berisi fungsi f yang menerima input matriks m berukuran n x (n+1). Kemudian program akan membagi matriks menjadi 2 buah matriks A berukuran n x n dan matriks B berukuran n x 1. Solusi SPL x_i didapat dengan cara mengganti kolom ke i matriks A dengan matriks B kemudian membagi determinan matriks A yang baru dengan determinan matriks A yang original. Dalam prosesnya juga akan menampilkan hasil SPL di layar dan memberi pilihan apakah user ingin menyimpan jawaban dalam file .txt.

3.5. Menghitung Determinan (Determinan.java)

File ini berisi 2 fungsi, yaitu detCof dan detObe masing masing mencari determinan dengan cara ekspansi kofaktor dan dengan cara reduksi baris. Cara ekspansi kofaktor dilakukan dengan pemanggilan fungsi detCof secara rekursif dengan basis jika matriks berukuran 1 x 1 yang akan mengembalikan nilai satu satunya elemen. detObe bekerja dengan melakukan OBE pada matriks tanpa perkalian baris dengan koefisien, hingga menjadi matriks segitiga atas. Determinan didapat dengan cara mengalikan semua angka pada diagonal utama dan dikali dengan $(-1)^p$ dimana p adalah banyaknya pertukaran baris selama OBE.

3.6. Menghitung Balikan Matriks (Inverse.java)

File ini berisi 2 fungsi, yaitu inverseCof dan inverseObe masing masing mencari Invers dari matriks persegi dengan cara ekspansi kofaktor dan dengan cara operasi baris elementer. Cara ekspansi kofaktor dilakukan dengan membuat sebuah matriks entri kofaktor dari matriks input dan dibagi dengan determinan matriks awal. Cara OBE didapat dengan cara menambahkan matriks identitas dengan ukuran sama di sebelah kanan matriks awal. Kemudian dilakukan eliminasi gauss jordan sehingga terdapat matriks identitas di kiri. Jika terbentuk sebuah matriks identitas, maka invers dari matriks input adalah matriks di sebelah kanan matriks identitas tersebut.

3.7. Melakukan Interpolasi Polinomial (PolynomialInterpolation.java)

File PolynomialInterpolation.java berisi fungsi f yang menerima input scanner. Kemudian fungsi akan menanyakan apakah user akan melakukan input titik menggunakan keyboard atau file. Jika menggunakan keyboard, fungsi akan menerima n yaitu jumlah titik yang akan diinterpolasi, titik dalam bentuk (x,y) sejumlah n, dan nilai x yang akan ditaksir. Jika titik yang diinput tidak sesuai dari segi jumlah atau format, atau jika ada titik yang muncul lebih dari sekali, input titik akan diulang. Jika x yang akan ditaksir di luar *range* x0...xn, maka input x akan diulang. Jika user memilih input menggunakan file, user akan menginput nama file yang berisi data titik dan x yang akan ditaksir. Jika titik ataupun x yang akan ditaksir tidak valid, fungsi akan berhenti dan user akan kembali ke main.

Titik-titik yang diinput tadi akan dimasukkan ke matriks berukuran n x n+1 dengan kolom berindeks 0 – n-1 di setiap baris akan berisi nilai x_{baris}^0 sampai x_{baris}^n, dan kolom dengan index n berisi nilai y_{baris}. Kemudian fungsi akan memanggil fungsi gauss untuk menyelesaikan SPL.

Fungsi akan mencetak hasil penyelesaian SPL dan perhitungan taksiran tersebut dengan format f(x) = a_nx^n + a_{n-1}x^n-1 + … + a_0, f(*x taksiran*) = …

Terakhir, fungsi akan menanyakan apakah user ingin menyimpan hasil tersebut dalam file .txt.

File ini juga memiliki fungsi findXn dan findX0 untuk validasi input x yang akan ditaksir, dan fungsi arePointsUnique (untuk matrix) dan arePointsUniqueS (untuk array of string) untuk validasi input titik.

3.8. Melakukan Bicubic Spline Interpolation (BiSplineInterpolation.java)

File BiSplineInterpolation.java berisi fungsi f yang menerima input scanner. Fungsi akan meminta user untuk memasukkan file yang berisi nilai f(0,0) sampai $f_{xy}(1,1)$ dan titik yang akan ditaksir nilainya, lalu menyimpan nilai f di sebuah matriks dan titik di dua variabel yang berbeda (satu untuk x dan satu untuk y). Kemudian, fungsi akan membuat matriks X dengan fungsi xMatrix lalu menginverse matriks tersebut dengan fungsi inverseObe. Selanjutnya, fungsi akan mengalikan invers matriks X dengan matriks berisi nilai f untuk mendapatkan matriks yang berisi nilai a. Fungsi akan menghitung nilai f(x,y) untuk titik yang akan ditaksir dengan rumus

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

Fungsi akan mencetak hasil perhitungan f(x,y) ke layar dengan format f(*x*,*y*) = …

Terakhir, fungsi akan menanyakan apakah user ingin menyimpan hasil tersebut dalam file .txt.

Selain fungsi f dan xMatrix yang sudah disebutkan, file juga berisi fungsi countResult untuk menghitung nilai taksiran dan readF untuk mengubah matriks hasil pembacaan readTXT ke bentuk 16x1 supaya bisa dikalikan dengan matriks X.

3.9. Regresi Linear Berganda (LinearReg.java)

Program ini akan meminta parameter Scanner dan sebuah matriks, lalu akan dibuat matriks baru yang berukuran kolom matriks sebelumnya x kolom matriks sebelumnya+1. Setelah itu, akan dilanjutkan sesuai rumus yang dijelaskan di Teori Singkat 2.6 tentang Regresi Linear Berganda. Program akhirnya menampilkan fungsi yang didapat, dan akan menanyakan apakah user akan menginput nilai untuk dihitung menggunakan fungsi yang didapat sebelumnya. Terakhir program akan menanyakan user apakah hasil-hasil di atas akan diwrite ke txt atau tidak.

Bab 4: Eksperimen

4.1. Percobaan Baca Tulis File

4.1.1. Percobaan Baca File

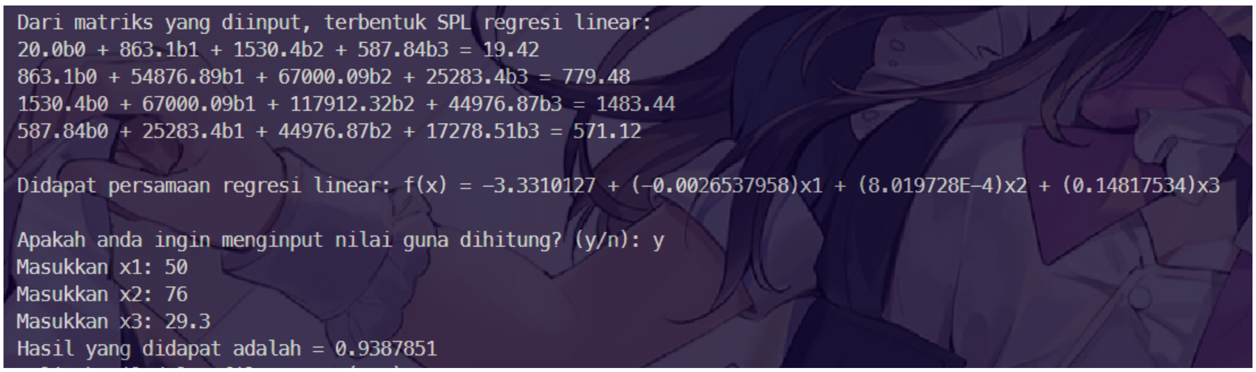
Fungsi yang digunakan untuk membaca file, readTXT, akan meminta input nama file (harus menggunakan extensionnya) yang akan dibaca. Fungsi akan mencari file dengan nama yang sesuai di dalam folder test.

Menggunakan data yang nantinya akan digunakan di Studi Kasus 6, penulis memilih input dari file bernama “linreg.txt”, dan perhitungan lebih lanjut sukses dilaksanakan seperti gambar berikut:

4.1.2. Percobaan Tulis File

Fungsi yang digunakan untuk menulis file, writeTXT, akan meminta input nama file (harus menggunakan extensionnya) yang akan dibaca. Jika file dengan nama tersebut ditemukan di dalam folder ./test (folder test di dalam src), fungsi akan membuka file tersebut dan menulis di file tersebut tanpa melakukan *overwrite*. Jika tidak, fungsi akan membuat file baru dengan nama yang diinput.

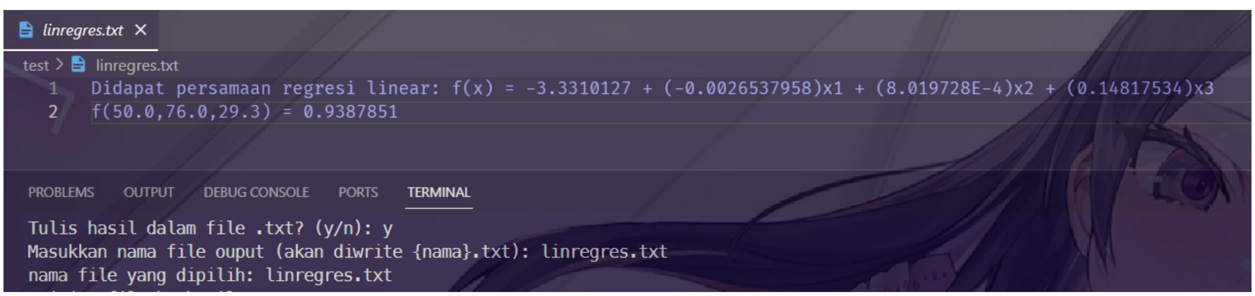
Menggunakan data dari percobaan Regresi Linear seperti berikut:


```
Dari matriks yang diinput, terbentuk SPL regresi linear:
20.0b0 + 863.1b1 + 1530.4b2 + 587.84b3 = 19.42
863.1b0 + 54876.89b1 + 67000.09b2 + 25283.4b3 = 779.48
1530.4b0 + 67000.09b1 + 117912.32b2 + 44976.87b3 = 1483.44
587.84b0 + 25283.4b1 + 44976.87b2 + 17278.51b3 = 571.12

Didapat persamaan regresi linear: f(x) = -3.3310127 + (-0.0026537958)x1 + (8.019728E-4)x2 + (0.14817534)x3

Apakah anda ingin menginput nilai guna dihitung? (y/n): y
Masukkan x1: 50
Masukkan x2: 76
Masukkan x3: 29.3
Hasil yang didapat adalah = 0.9387851
```

Setelah itu, hasil persamaan dan hasil aproksimasinya, dapat ditulis di file seperti berikut:


```
linregres.txt
test > linregres.txt
1   Didapat persamaan regresi linear: f(x) = -3.3310127 + (-0.0026537958)x1 + (8.019728E-4)x2 + (0.14817534)x3
2   f(50.0,76.0,29.3) = 0.9387851

PROBLEMS   OUTPUT   DEBUG CONSOLE   PORTS   TERMINAL

Tulis hasil dalam file .txt? (y/n): y
Masukkan nama file ouput (akan diwrite {nama}.txt): linregres.txt
nama file yang dipilih: linregres.txt
```

4.2. Studi Kasus 1: Sistem Persamaan Linear Ax = b

4.2.1. Studi Kasus 1 Bagian A

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

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 1
Opsi 1 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk in
put dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
1
Masukkan ukuran baris dari matriks A: 4
Masukkan ukuran kolom dari matriks A: 4
Masukkan matriks A:
1 1 -1 -1
2 5 -7 -5
2 -1 1 3
5 2 -4 2
Masukkan matriks B (ukuran baris x 1) dalam bentuk baris:
1
-2
4
6
Tidak dapat mencari solusi SPL.
```

SPL diselesaikan dengan metode Gauss. Tidak dapat mencari SPL karena terdapat baris yang koefisien semua peubahnya adalah 0 dan memiliki solusi yang bukan 0.

4.2.2. Studi Kasus 1 Bagian B

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

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 1
Opsi 1 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk input dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
1
Masukkan ukuran baris dari matriks A: 4
Masukkan ukuran kolom dari matriks A: 5
Masukkan matriks A:
1 -1 0 0 1
1 1 0 -3 0
2 -1 0 1 -1
-1 2 0 -2 -1
Masukkan matriks B (ukuran baris x 1) dalam bentuk baris:
3
6
5
-1
Hasil perhitungan menggunakan metode Gauss :
x1 : 3.0-(1.0*(t))+ (-1.0*(-1.0-(-3.0*(t))+ (1.0*(-1.0-(-1.0*(t))))))
x2 : -1.0-(-3.0*(t))+ (1.0*(-1.0-(-1.0*(t))))
x3 : free
x4 : -1.0-(-1.0*(t))
x5 : t
```

SPL diselesaikan dengan metode Gauss.

4.2.3. Studi Kasus 1 Bagian C

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

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 1
Opsi 1 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk input dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
1
Masukkan ukuran baris dari matriks A: 3
Masukkan ukuran kolom dari matriks A: 6
Masukkan matriks A:
0 1 0 0 1 0
0 0 0 1 1 0
0 1 0 0 0 1
Masukkan matriks B (ukuran baris x 1) dalam bentuk baris:
2
-1
1
Hasil perhitungan menggunakan metode Gauss :
x1 : free
x2 : 2.0-(1.0*(1.0-(-1.0*(t))))
x3 : free
x4 : -1.0-(1.0*(1.0-(-1.0*(t))))
x5 : 1.0-(-1.0*(t))
x6 : t
```

SPL diselesaikan dengan metode Gauss.

4.2.4. Studi Kasus 1 Bagian D

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

SPL diselesaikan dengan metode Cramer.

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 4
Opsi 4 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk in
put dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
2
Masukkan ukuran baris dari matriks: 6
Masukkan ukuran kolom dari matriks: 7
Masukkan matriks:
1 0.5 0.33 0.23 0.20 0.17 1
0.5 0.33 0.25 0.20 0.17 0.14 0
0.33 0.25 0.20 0.17 0.14 0.13 0
0.25 0.20 0.17 0.14 0.13 0.11 0
0.20 0.17 0.14 0.13 0.11 0.10 0
0.17 0.14 0.13 0.11 0.10 0.09 0
Hasil perhitungan menggunakan metode Cramer :
x1 : 2.8572025
x2 : 0.5573829
x3 : -10.592542
x4 : -10.452899
x5 : 5.0173335
x6 : 16.236889
```

Untuk n = 6

```
Silahkan pilih metode atau pilihan menu: 4
Opsi 4 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk in
put dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
2
Masukkan ukuran baris dari matriks: 10
Masukkan ukuran kolom dari matriks: 11
Masukkan matriks:
1 0.5 0.33 0.23 0.20 0.17 0.14 0.13 0.11 0.10 1
0.5 0.33 0.25 0.20 0.17 0.14 0.13 0.11 0.10 0.09 0
0.33 0.25 0.20 0.17 0.14 0.13 0.11 0.10 0.09 0.08 0
0.25 0.20 0.17 0.14 0.13 0.11 0.10 0.09 0.08 0.08 0
0.20 0.17 0.14 0.13 0.11 0.10 0.09 0.08 0.08 0.07 0
0.17 0.14 0.13 0.11 0.10 0.09 0.08 0.08 0.07 0.07 0
0.14 0.13 0.11 0.10 0.09 0.08 0.08 0.07 0.07 0.06 0
0.13 0.11 0.10 0.09 0.08 0.08 0.07 0.07 0.06 0.06 0
0.11 0.10 0.09 0.08 0.08 0.07 0.07 0.06 0.06 0.06 0
0.10 0.09 0.08 0.08 0.07 0.07 0.06 0.06 0.06 0.05 0
Hasil perhitungan menggunakan metode Cramer :
x1 : -6.9736576
x2 : 37.197876
x3 : -16.276594
x4 : -18.599974
x5 : -6.9679236
x6 : -4.648219
x7 : -37.206123
x8 : 16.274197
x9 : 11.624584
x10 : 30.230703
```

Untuk n = 10

4.3. Studi Kasus 2: SPL Berbentuk Matriks Augmented

4.3.1 Studi Kasus 2 Bagian A

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

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 1
Opsi 1 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk input dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
2
Masukkan ukuran baris dari matriks: 4
Masukkan ukuran kolom dari matriks: 5
Masukkan matriks:
1 -1 2 -1 -1
2 1 -2 -2 -2
-1 2 -4 1 1
3 0 0 -3 -3
Hasil perhitungan menggunakan metode Gauss :
x1 : -1.0-(-1.0*(s))+ (2.0*(t))+ (-1.0*(0.0-(-2.0*(t))))
x2 : 0.0-(-2.0*(t))
x3 : t
x4 : s
```

SPL diselesaikan dengan metode Gauss.

4.3.2 Studi Kasus 2 Bagian B

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

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 1
Opsi 1 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk in
put dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
2
Masukkan ukuran baris dari matriks: 6
Masukkan ukuran kolom dari matriks: 5
Masukkan matriks:
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
Hasil perhitungan menggunakan metode Gauss :
x1 : 4.0
x2 : 6.0
x3 : 1.0
x4 : 1.0
```

SPL diselesaikan dengan metode Gauss.

4.4. Studi Kasus 3: SPL Berbentuk

4.4.1 Studi Kasus 3 Bagian A

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

Persamaan di atas, diubah menjadi sebuah matriks *augmented* seperti berikut:

```
spllumayanbrutal.txt
8 1 3 2 0
2 9 -1 -2 1
1 3 2 -1 2
1 0 6 4 3
```

Setelah dimasukkan ke program, dengan menggunakan metode Cramer, diperoleh hasil x1 = -0.22432433 ; x2 = 0.18243243 ; x3 = 0.7094595 ; x4 = -0.2581081 seperti gambar di bawah:

```
nama file yang dipilih: spllumayanbrutal.txt
Hasil perhitungan menggunakan metode Cramer :
x1 : -0.22432433
x2 : 0.18243243
x3 : 0.7094595
x4 : -0.2581081
```

4.4.2 Studi Kasus 3 Bagian B

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

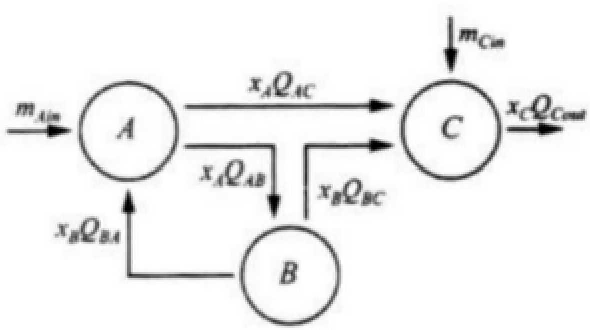
Persamaan berikut diubah menjadi sebuah matriks augmented sebagai berikut:

```
test > splbrutal.txt
0 0 0 0 0 0 1 1 1 13.00
0 0 0 1 1 1 0 0 0 15.00
1 1 1 0 0 0 0 0 0 8.00
0 0 0.04289 0 0.04289 0.75 0.04289 0.75 0.61396 14.79
0 0.25 0.91421 0.24 0.91421 0.24 0.91421 0.24 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 18.00
0 1 0 0 1 0 0 1 0 12.00
1 0 0 1 0 0 1 0 0 6.00
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
```

Lalu kemudian matriks tersebut dimasukkan ke kalkulator, menunjukkan bahwa tidak dapat mencari solusi dari SPL di atas (tidak ada jawaban).

```
Masukkan nama file: splbrutal.txt
nama file yang dipilih: splbrutal.txt
Tidak dapat mencari solusi SPL.
```

4.5. Studi Kasus 4: Sistem Reaktor



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

$$
\begin{aligned}
&\text{A:} \quad m_{A_{in}} + Q_{BA}x_B - Q_{AB}x_A - Q_{AC}x_A = 0 \\
&\text{B:} \quad Q_{AB}x_A - Q_{BA}x_B - Q_{BC}x_B = 0 \\
&\text{C:} \quad m_{C_{in}} + Q_{AC}x_A + Q_{BC}x_B - Q_{C_{out}}x_C = 0
\end{aligned}
$$

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.

Jika permasalahan tersebut diubah menjadi SPL akan menjadi seperti ini

```
SISTEM PERSAMAAN LINEAR
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah cramer
5. Kembali
Silahkan pilih metode atau pilihan menu: 4
Opsi 4 telah dipilih.
Silahkan Input Matrix yang akan diolah
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk in
put dari file.txt
1
Masukkan pilihan input:
(1) Format Ax = B
(2) Format Matriks Augmented
2
Masukkan ukuran baris dari matriks: 3
Masukkan ukuran kolom dari matriks: 4
Masukkan matriks:
-120 60 0 -1300
40 -80 0 0
80 20 -150 -200
Hasil perhitungan menggunakan metode Cramer :
x1 : 14.444445
x2 : 7.2222223
x3 : 10.0
```

SPL tersebut diselesaikan dengan metode Cramer

4.6. Studi Kasus 5: Interpolasi Polinomial

4.6.1. Studi Kasus 5 Bagian A

Dengan menggunakan titik-titik sesuai tabel berikut dengan urutan yang sama:

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

Diperoleh fungsi f(x) sebagai berikut

```
f(x) = (-1.9674693069285452E-14)x^6 + (8.37461474950599E-14)x^5 + (0.02604166666652843)x^4 + (1.1103618025032347E-13)x^3 + (0.1973958
3333328886)x^2 + (0.24000000000000804)x^1 + -0.02297656250000046, f(0.2) = 0.032960937500000065
```

$f(x) = (-1.9674693069285452E\text{-}14)x^6 + (8.37461474950599E\text{-}14)x^5 + (0.02604166666652843)x^4 + (1.1103618025032347E\text{-}13)x^3 + (0.19739583333328886)x^2 + (0.24000000000000804)x + -0.02297656250000046$

Maka didapatkan:
f(0.2) = 0.032960937500000065
f(0.55) = 0.17111865234375
f(0.85) = 0.33723583984375005
f(1.28) = 0.6775418375

Semua titik dapat dicari nilainya karena berada dalam rentang [0.1…1.3].

4.6.2. Studi Kasus 5 Bagian B

Dengan menggunakan titik-titik sesuai tabel berikut (koma di kolom tanggal diganti titik dan titik penanda ribuan di kolom kasus dihilangkan):

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

Diperoleh fungsi f(x) sebagai berikut

```
f(x) = (-141006.35265450666)x^9 + (9373741.525538526)x^8 + (-2.7550250306447875E8)x^7 + (4.696316966551336E9)x^6 + (-5.11378648581328
1E10)x^5 + (3.685975674434824E11)x^4 + (-1.757053341532463E12)x^3 + (5.335014967217977E12)x^2 + (-9.348572690402438E12)x^1 + 7.188430
348204094E12, f(8.543) = 13936.265625
```

$$f(x) = (-141006.35265450666)x^9 + (9373741.525538526)x^8 + (-2.7550250306447875E8)x^7 + (4.696316966551336E9)x^6 + (-5.113786485813281E10)x^5 + (3.685975674434824E11)x^4 + (-1.757053341532463E12)x^3 + (5.335014967217977E12)x^2 + (-9.348572690402438E12)x + 7.188430348204094E12$$

Maka, untuk tanggal-tanggal berikut didapatkan taksiran jumlah kasus baru sebesar:
16/07/2022 : f(7.516) = 53537.625
10/08/2022 : f(8.322) = 36343.890625
05/09/2022 : f(9.167) = -
17/08/2022 : f(8.543) = 13936.265625

Untuk tanggal 5 September, fungsi diprogram untuk mengeluarkan *error message* karena x = 9.167 berada di luar range $[x_0...x_n]$ = [6.567…9]. Jika dihitung akan menghasilkan bilangan negatif, sebuah angka yang tidak mungkin untuk persoalan yang sedang diinterpolasi (jumlah kasus COVID).

Hal ini terjadi karena Runge’s Phenomenon, salah satu kelemahan interpolasi polinomial. Runge’s Phenomenon adalah osilasi di kedua ujung grafik hasil interpolasi polinomial berderajat tinggi.

4.6.3. Studi Kasus 5 Bagian C

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

Untuk polinom berderajat n, x yang diambil berada di selang [0,2] dengan jarak 2/n. Misalkan untuk n=2,

f(0) = 0
f(1) = 0.53788
f(2) = 0.57665

Titik-titiknya adalah (0,0), (1,0.53788), (2,0.57665)

Untuk n=4,
f(0) = 0
f(0,5) = 0.44543
f(1) = 0.53788
f(1,5) = 0.58090
f(2) = 0.57665

Titik-titiknya adalah (0,0), (0.5,0.44543), (1,0.53788), (1.5,0.58090), (2,0.57665)

Untuk n=10,
f(0) = 0

f(0.2) = 0.34277
f(0.4) = 0.41888
f(0.6) = 0.46843
f(0.8) = 0.50716
f(1) = 0.53788
f(1.2) = 0.56092
f(1.4) = 0.57619
f(1.6) = 0.58368
f(1.8) = 0.58367
f(2) = 0.57665

Titik-titiknya adalah (0,0), (0.2,0.34277), (0.4, 0.41888), (0.6,0.46843), (0.8,0.50716), (1,0.53788), (1.2,0.56092), (1.4, 0.57619), (1.6,0.58368), (1.8,0.58367), (2,0.57665)

Dengan menyulihkan titik-titik tersebut ke program (dengan x yang akan ditaksir sembarang), akan didapatkan persamaan:

```
Menu Interpolasi Polinom telah dipilih
Masukkan pilihan input: (1) untuk input dari keyboard, dan (2) untuk input dari file.txt
1
Masukkan jumlah titik yang akan diinterpolasi (minimal 2):
3
Masukkan 3 titik:
(Format: (x0,y0), (x1,y1), ..., (xn,yn))
(0,0), (1,0.53788), (2,0.57665)
Masukkan nilai x yang akan ditaksir nilai fungsinya:
1.2
f(x) = (-0.24955500000000003)x^2 + (0.7874350000000001)x^1 + 0.0, f(1.2) = 0.5855627999999999
Tulis hasil dalam file .txt? (y/n): y
Masukkan nama file ouput (akan diwrite {nama}.txt): SKIPc.txt
nama file yang dipilih: SKIPc.txt
Writing file berhasil.
```

$f(x) = (-0.24955500000000003)x^2 + (0.7874350000000001)x + 0.0$

$f(x) = (-0.20092666666666617)x^4 + (1.0075133333333315)x^3 + (-1.8656083333333313)x^2 + (1.596901666666666)x + 0.0$

$f(x) = (-0.47339922108775473)x^{10} + (5.230481451508182)x^9 + (-25.25715499679231)x^8 + (70.03480617964688)x^7 + (-123.20037886143578)x^6 + (143.34300762414784)x^5 + (-111.53709393881718)x^4 + (57.3305043320583)x^3 + (-18.814652966037244)x^2 + (3.8817603968090566)x + 0.0$

Untuk n=2, n=4, dan n=10, berturut-turut. Ketiga persamaan tersebut adalah penyederhanaan fungsi f(x) di awal. Semakin besar n, penyederhanaan fungsi f(x) akan semakin akurat karena dengan semakin banyak titik yang diinterpolasi, semakin besar kemungkinan grafik polinomial mendekati bentuk grafik fungsi aslinya.

4.7. Studi Kasus 6 : Regresi Linear Berganda

Menggunakan sekumpulan data sesuai pada tabel berikut dengan urutan yang sama:

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.

Diperoleh Model regresi terbentuk:

```
Dari matriks yang diinput, terbentuk SPL regresi linear:
20.0b0 + 863.1b1 + 1530.4b2 + 587.84b3 = 19.42
863.1b0 + 54876.89b1 + 67000.09b2 + 25283.4b3 = 779.48
1530.4b0 + 67000.09b1 + 117912.32b2 + 44976.87b3 = 1483.44
587.84b0 + 25283.4b1 + 44976.87b2 + 17278.51b3 = 571.12

Didapat persamaan regresi linear: f(x) = -3.3310127 + (-0.0026537958)x1 + (8.019728E-4)x2 + (0.14817534)x3
```

Demikian untuk humiditas 50% (x1), temperatur 76 deg F (x2), dan tekanan 29.3 (x3), didapat perkiraan N) sekitar 0.94 (hasil pembulatan 2 angka dibelakang koma), seperti berikut:

```
Apakah anda ingin menginput nilai guna dihitung? (y/n): y
Masukkan x1: 50
Masukkan x2: 76
Masukkan x3: 29.3
Hasil yang didapat adalah = 0.94
```

4.8. Studi Kasus 7: Interpolasi *Bicubic Spline*

Dengan matriks input yang berisi nilai f(0,0) sampai $f_{xy}(1,1)$ sebagai berikut:

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

Didapatkan nilai-nilai berikut:

```
Menu Interpolasi Bicubic Spine telah dipilih
Masukkan nama file: SKBSI.txt
nama file yang dipilih: SKBSI.txt
f(0.1,0.9) = 128.57518
Tulis hasil dalam file .txt? (y/n): y
Masukkan nama file ouput (akan diwrite {nama}.txt): SKBSIJ.txt
nama file yang dipilih: SKBSIJ.txt
Writing file berhasil.
```

```
src > test > SKBSIJ.txt
1    f(0.0,0.0) = 21.0
2    f(0.5,0.5) = 87.796875
3    f(0.25,0.75) = 117.73218
4    f(0.1,0.9) = 128.57518
```

f(0.0,0.0) = 21.0
f(0.5,0.5) = 87.796875
f(0.25,0.75) = 117.73218
f(0.1,0.9) = 128.57518

Semua data valid karena di dalam range [0…1] untuk x dan y.

Bab 5: Kesimpulan

5.1. Kesimpulan

Pada tugas besar ini, penulis telah mengimplementasikan

1. Matriks sebagai sebuah tipe data abstrak
2. Operasi pada matriks seperti transpos, perkalian, dan operasi baris elementer
3. Penyelesaian SPL menggunakan matriks dan metode eliminasi Gauss, eliminasi Gauss-Jordan, matriks balikan, dan Cramer
4. Perhitungan determinan matriks dengan metode reduksi baris dan ekspansi kofaktor
5. Penentuan matriks balikan dengan metode reduksi baris dan ekspansi kofaktor
6. Interpolasi polinomial dan *bicubic spline*
7. Perhitungan regresi linear berganda

Hal-hal tersebut diimplementasikan dengan membuat pustaka matriks dengan bahasa pemrograman Java yang berparadigma *Object Oriented Programming*, sehingga kami dapat memanfaatkan fungsi-fungsi dari pustaka dan *class* lain.

5.2. Saran dan Komentar

Penulis menyarankan untuk lebih mendalami bahasa pemrograman Java untuk percobaan berikutnya. Ada beberapa masalah yang penulis hadapi karena belum terlalu mengenali Java, dan mungkin algoritma yang digunakan bisa menjadi lebih sederhana menggunakan fungsi dari pustaka Java yang belum penulis ketahui.

Selain itu, sebaiknya struktur semua *class* dipikirkan dari awal supaya program bisa dibuat lebih modular dan mengurangi fungsi yang memiliki dua versi seperti pada file Gauss.java kami (ada dua fungsi Gauss, salah satunya adalah versi yang tidak mencetak teks supaya bisa digunakan di fungsi lain, sedangkan yang satunya adalah versi yang dijalankan ketika memilih menu SPL Gauss, dan akan mencetak teks hasil ke layar).

Penulis juga sangat menyarankan untuk memulai pekerjaan dari jauh hari supaya tidak terburu-buru dalam penyelesaian tugas besar ini.

5.3 Refleksi dan Komentar

Melalui tugas besar ini, penulis menyadari bahwa matriks dapat diimplementasikan untuk menyelesaikan berbagai macam masalah, seperti penyelesaian banyak SPL sekaligus, interpolasi, dan regresi linear dengan cepat. Matriks sebagai tipe data juga lumayan mudah untuk digunakan.

Penulis juga menyadari bahwa ada banyak cara yang berbeda untuk menyelesaikan permasalahan matriks. Setiap metode memiliki kelebihan dan kekurangannya sendiri, baik dari segi kerumitan algoritma, kecepatan pelaksanaan, dan lainnya.

Daftar Referensi

- Modul Tugas Besar I IF2123 Aljabar Linear dan Geometri
- Bahan pembelajaran mata kuliah IF2123 Aljabar Linear dan Geometri 2023/2024, oleh Rinaldi Munir
- https://engcourses-uofa.ca/introduction-to-numerical-analysis/polynomial-interpolation/
- https://en.wikipedia.org/wiki/Runge%27s_phenomenon