

**Tugas Besar 1 IF 2123 Aljabar Linier dan Geometri**  
**Sistem Persamaan Linier, Determinan, dan Aplikasinya**



Kelompok 40 - IntVerse  
Hosea Nathanael Abetnego - 13521057  
Ahmad Ghulam Ilham - 13521118  
Muhammad Naufal Nalendra - 13521152  
**Semester I Tahun 2022/2023**

## **BAB 1**

### **Deskripsi Masalah**

Sistem persamaan linier (SPL) banyak ditemukan di dalam bidang sains dan rekayasa. Sudah dipelajari berbagai metode untuk menyelesaikan SPL, termasuk menghitung determinan matriks. Sembarang SPL dapat diselesaikan dengan beberapa metode, yaitu metode eliminasi Gauss, metode eliminasi Gauss-Jordan, metode matriks balikan ( $x = A^{-1}b$ ), dan kaidah Cramer (khusus untuk SPL dengan  $n$  peubah dan  $n$  persamaan). Solusi sebuah SPL mungkin tidak ada, banyak (tidak berhingga), atau hanya satu (unik/tunggal).

Di dalam Tugas Besar 1 ini, akan dibuat satu atau lebih library aljabar linier dalam Bahasa Java. Library tersebut berisi fungsi-fungsi seperti eliminasi Gauss, eliminasi Gauss-Jordan, menentukan balikan matriks, menghitung determinan, kaidah Cramer (kaidah Cramer khusus untuk SPL dengan  $n$  peubah dan  $n$  persamaan). Selanjutnya, gunakan library tersebut di dalam program Java untuk menyelesaikan berbagai persoalan yang dimodelkan dalam bentuk SPL, menyelesaikan persoalan interpolasi, dan persoalan regresi.



Bentuk matriks *augmented* dari SPL tersebut adalah :

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

Lakukan OBE pada matriks hingga terbentuk Matriks Eselon Baris :

$$\begin{bmatrix} 1 & 3/2 & -1/2 & 5/2 \\ 0 & 1 & 1/2 & 7/2 \\ 0 & 0 & 1 & 3 \end{bmatrix}$$

Diperoleh persamaan-persamaan linier sebagai berikut :

$$x_1 + 3/2x_2 - 1/2x_3 = 5/2 \quad (i)$$

$$x_2 + 1/2x_3 = 7/2 \quad (ii)$$

$$x_3 = 3 \quad (iii)$$

Dengan dilakukan penyulihan mundur, didapatkan bahwa  $x_1 = 1$ ,  $x_2 = 2$ ,  $x_3 = 3$ .

## II. Metode Eliminasi Gauss-Jordan

Metode ini merupakan pengembangan dari Metode Eliminasi Gauss. Metode Eliminasi Gauss-Jordan menggunakan prinsip yang sama dengan Metode Eliminasi Gauss, tetapi melakukan penyederhanaan lebih lanjut sehingga terbentuk Matriks Eselon Baris Tereduksi. Contoh :

Terdapat persamaan linier sebagai berikut :

$$x_1 - x_2 + 2x_3 - x_4 = -1$$

$$2x_1 + x_2 - 2x_3 - 2x_4 = -2$$

$$-x_1 + 2x_2 - 4x_3 + x_4 = 1$$

$$3x_1 - 3x_4 = -3$$

Ubah persamaan tersebut menjadi matriks augmented :

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

Lakukan OBE terhadap matriks tersebut hingga terbentuk Matriks Eselon Baris Tereduksi :

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

Diperoleh persamaan sebagai berikut :

$$x_1 - x_4 = -1 \quad (\text{i})$$

$$x_2 - 2x_3 = 0 \quad (\text{ii})$$

Dari persamaan tersebut didapatkan bahwa solusi SPL tidak unik (memiliki banyak solusi), sehingga  $x_1 = s - 1$ ,  $x_2 = 2r$ ,  $x_3 = r$ ,  $x_4 = s$ .

### III. Determinan

Determinan merupakan faktor penskalaan transformasi pada sebuah matriks. Determinan biasa dilambangkan dengan  $\det(A)$  atau  $|A|$  dan determinan hanya bisa didapat pada matriks persegi/simetri yaitu matriks yang baris dan kolomnya sama banyak. Pada tugas besar ini, diterapkan dua cara untuk mencari determinan sebuah matriks yaitu dengan Metode Reduksi Baris dan Ekspansi Kofaktor.

Metode Reduksi Baris dilakukan dengan dilakukan OBE pada matriks sehingga terbentuk matriks segitiga atas atau bawah. Seperti berikut :

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \xrightarrow{\text{OBE}} \begin{bmatrix} a'_{11} & a'_{12} & \dots & a'_{1n} \\ 0 & a'_{22} & \dots & a'_{2n} \\ \vdots & \vdots & \dots & a'_{3n} \\ 0 & 0 & 0 & a'_{nn} \end{bmatrix}$$

(Contoh segitiga atas)

Setelah didapatkan matriks segitiga, determinan dapat diperoleh dengan mengalikan diagonal utama matriks tersebut sehingga  $\det(A)$  adalah :

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

Catatan : p adalah jumlah pertukaran baris yang dilakukan pada OBE.

Metode selanjutnya adalah Ekspansi Kofaktor. Metode Kofaktor dapat dilakukan dengan mengalikan tiap elemen matriks utama dengan kofaktornya dalam satu baris atau kolom. Contoh :

$$A = \begin{bmatrix} 3 & -1 & 2 \\ 5 & 0 & 4 \\ 8 & 2 & -3 \end{bmatrix}$$

Maka  $\det(A)$  adalah :

$$\det(A) = a_{11}C_{11} + a_{12}C_{12} + a_{13}C_{13}$$

$$\det(A) = 3 \begin{vmatrix} 0 & 4 \\ 2 & 3 \end{vmatrix} - (-1) \begin{vmatrix} 5 & 4 \\ 8 & -3 \end{vmatrix} + 2 \begin{vmatrix} 5 & 0 \\ 8 & 2 \end{vmatrix}$$

#### IV. Matriks Balikan

Matriks Balikan atau invers matriks adalah, seperti namanya, balikan dari sebuah Matriks. Matriks balikan dapat diperoleh hanya jika matriks utama merupakan matriks persegi. Jika perkalian matriks  $AB = BA = I$ , maka B merupakan invers matriks dari matriks A. Terdapat dua metode yang bisa digunakan untuk mencari balikan dari sebuah matriks. Metode pertama adalah dengan Eliminasi Gauss-Jordan dan Matriks Adjoin.

Metode Eliminasi Gauss-Jordan dilakukan dengan melakukan OBE pada matriks gabungan dari matriks utama dengan matriks identitas.

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

(G-J merupakan singkatan dari Gauss-Jordan)

Matriks yang berada di sisi kanan matriks baris tereduksi (matriks identitas) adalah invers dari matriks utama. Contoh :

Terdapat matriks sebagai berikut,

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 5 & 3 \\ 1 & 0 & 8 \end{bmatrix}$$

Gabungkan matriks dengan matriks identitas,

$$\left( \begin{array}{ccc|ccc} 1 & 2 & 3 & 1 & 0 & 0 \\ 2 & 5 & 3 & 0 & 1 & 0 \\ 1 & 0 & 8 & 0 & 0 & 1 \end{array} \right)$$

Lakukan OBE pada matriks tersebut sehingga membentuk matriks baris tereduksi pada sisi kiri matriks,

$$\left( \begin{array}{ccc|ccc} 1 & 0 & 0 & -40 & 16 & 9 \\ 0 & 1 & 0 & 13 & -5 & -3 \\ 0 & 0 & 1 & 5 & -2 & -1 \end{array} \right)$$

Maka invers matriks adalah

$$A^{-1} = \begin{bmatrix} -40 & 16 & 9 \\ 13 & -5 & -3 \\ 5 & -2 & -1 \end{bmatrix}$$

Metode kedua adalah Matriks Adjoin. Metode ini dilakukan dengan mengalikan matriks kofaktor dari matriks utama dengan  $1/\det$  dari matriks utama seperti berikut,

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

Contoh, terdapat matriks sebagai berikut

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

Setelah dihitung,  $\det(A) = 64$ , kemudian tentukan matriks kofaktor. Matriks adjoin merupakan *transpose* matriks dari matriks kofaktor sehingga didapat matriks adjoin adalah sebagai berikut

$$\begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & -10 & 16 \end{bmatrix}$$

Maka, invers matriks yang didapatkan adalah

$$\frac{1}{64} \begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & -10 & 16 \end{bmatrix}$$

$$\begin{bmatrix} 12/64 & 4/64 & 12/64 \\ 6/64 & 2/64 & -10/64 \\ -16/64 & -10/64 & 16/64 \end{bmatrix}$$

#### V. Matriks Kofaktor

Matriks Kofaktor merupakan matriks yang elemen-elemennya merupakan kofaktor dari matriks utama. Matriks kofaktor didapatkan dengan menghitung determinan matriks minor dari elemen yang ingin dicari, dikalikan dengan sebuah faktor (- atau +). Matriks minor atau upa-matriks merupakan matriks yang elemennya merupakan elemen pada matriks utama yang tidak terdapat pada baris atau kolom tertentu. Jika  $M_{ij}$  adalah determinan matriks minor, maka nilai determinan tersebut didapatkan dari determinan upa-matriks yang elemen-elemennya tidak terdapat di baris  $i$  atau kolom  $j$  pada matriks utama. Seperti berikut

$$A = \begin{bmatrix} 6 & -3 & 1 \\ 2 & 2 & -4 \\ 1 & 5 & 3 \end{bmatrix} \quad M_{11} = \begin{vmatrix} 2 & -4 \\ 5 & 3 \end{vmatrix} = (2)(3) - (-4)(5) = 26$$

Kemudian, determinan tersebut dikalikan dengan faktor dari matriks sehingga  $C_{ij}$  adalah elemen matriks kofaktor (kofaktor entri  $a_{ij}$ ) pada baris  $i$ , kolom  $j$  dan, nilai  $C_{ij}$  adalah

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

Terdapat sebuah matriks sebagai berikut

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



Maka Matriks Kofaktor dari matriks tersebut adalah

$$\begin{bmatrix} + \begin{vmatrix} 6 & 3 \\ 4 & 0 \end{vmatrix} & - \begin{vmatrix} 1 & 3 \\ 2 & 0 \end{vmatrix} & + \begin{vmatrix} 1 & 6 \\ 2 & -4 \end{vmatrix} \\ - \begin{vmatrix} 2 & -1 \\ -4 & 0 \end{vmatrix} & + \begin{vmatrix} 3 & -1 \\ 2 & 0 \end{vmatrix} & - \begin{vmatrix} 3 & 2 \\ 2 & -4 \end{vmatrix} \\ + \begin{vmatrix} 2 & -1 \\ 6 & 3 \end{vmatrix} & - \begin{vmatrix} 3 & -1 \\ 1 & 3 \end{vmatrix} & + \begin{vmatrix} 3 & 2 \\ 1 & 6 \end{vmatrix} \end{bmatrix} = \begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix}$$

## VI. Matriks Adjoin

Matriks Adjoin merupakan hasil transpose dari matriks kofaktor. Matriks adjoin digunakan untuk mencari invers dari sebuah matriks dengan mengalikannya dengan  $1/\det$  dari matriks yang ingin ditentukan balikkannya.

Contoh, terdapat sebuah matriks

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

Matriks kofaktor dari matriks tersebut adalah

$$\begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix}$$

Sehingga matriks adjoin dari matriks tersebut adalah

$$\text{adj}(A) = \begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & -10 & 16 \end{bmatrix}$$

## VII. Kaidah Cramer

Kaidah Cramer memanfaatkan sebuah formula untuk menyelesaikan sebuah SPL dengan menggunakan determinan matriks *augmented*. Jika terdapat  $\mathbf{Ax} = \mathbf{b}$  merupakan SPL dengan  $n$  variabel dan  $\det(A) \neq 0$ , maka solusi SPL tersebut dengan Kaidah Cramer adalah

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

Dengan  $A_n$  adalah matriks  $A$  dengan kolom  $n$  diganti dengan matriks  $\mathbf{b}$ .

Terdapat sistem persamaan linier sebagai berikut

$$-x + 2y - 3z = 1$$

$$2x + z = 0$$

$$3x - 4y + 4z = 2$$

Maka SPL tersebut dalam matriks adalah

$$A = \begin{bmatrix} -1 & 2 & -3 \\ 2 & 0 & 1 \\ 3 & -4 & 4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

Diperoleh  $\det(A) = 10$ , kemudian tentukan  $\det(A_1)$ ,  $\det(A_2)$ , dan  $\det(A_3)$ , yaitu

$$A_1 = \begin{bmatrix} \mathbf{1} & 2 & -3 \\ \mathbf{0} & 0 & 1 \\ \mathbf{2} & -4 & 4 \end{bmatrix} \Rightarrow \det(A_1) = \begin{vmatrix} 1 & 2 & -3 \\ 0 & 6 & 1 \\ 2 & -4 & 4 \end{vmatrix} = 8$$

$$A_2 = \begin{bmatrix} -1 & \mathbf{1} & -3 \\ 2 & \mathbf{0} & 1 \\ 3 & \mathbf{2} & 4 \end{bmatrix} \Rightarrow \det(A_2) = \begin{vmatrix} -1 & 1 & -3 \\ 2 & 0 & 1 \\ 3 & 2 & 4 \end{vmatrix} = -15$$

$$A_3 = \begin{bmatrix} -1 & 2 & \mathbf{1} \\ 2 & 0 & \mathbf{0} \\ 3 & -4 & \mathbf{2} \end{bmatrix} \Rightarrow \det(A_3) = \begin{vmatrix} -1 & 2 & 1 \\ 2 & 0 & 0 \\ 3 & -4 & 2 \end{vmatrix} = 16$$

Maka nilai  $x_1$ ,  $x_2$ , dan  $x_3$  adalah

$$x_1 = \frac{\det(A_1)}{\det(A)} = \frac{8}{10} = \frac{4}{5} \quad x_2 = \frac{\det(A_2)}{\det(A)} = \frac{-15}{10} = -\frac{3}{2} \quad x_3 = \frac{\det(A_3)}{\det(A)} = \frac{16}{10} = \frac{8}{5}$$

## VIII. Interpolasi Polinom

Interpolasi Polinom adalah teknik interpolasi dengan asumsi bahwa data yang dimiliki mengikuti pola polinomial. Teknik ini digunakan untuk menentukan titik-titik diantara  $n$  buah titik menggunakan pendekatan fungsi polinomial pangkat  $n$ . Misal terdapat  $n+1$  buah titik yaitu  $(x_0, y_0), (x_1, y_1), (x_2, y_2) \dots, (x_n, y_n)$ , maka diperoleh persamaan

$$a_0 + a_1 x_0 + a_2 x_0^2 + \dots + a_n x_0^n = y_0$$

$$a_0 + a_1 x_1 + a_2 x_1^2 + \dots + a_n x_1^n = y_1$$

...

...

$$a_0 + a_1 x_n + a_2 x_n^2 + \dots + a_n x_n^n = y_n$$

Lakukan metode-metode yang dapat digunakan untuk menyelesaikan SPL tersebut sehingga diperoleh sebuah fungsi

$$p_n(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$$

Dengan  $a_0, a_1, a_2, \dots, a_n$  adalah solusi SPL. Fungsi tersebut kemudian dapat digunakan untuk menaksir nilai  $y$  pada suatu  $x$ .

## IX. Interpolasi Bikubik

Interpolasi Bikubik merupakan teknik interpolasi pada data 2D umumnya digunakan dalam pembesaran citra yang merupakan pengembangan dari interpolasi linier dan kubik.

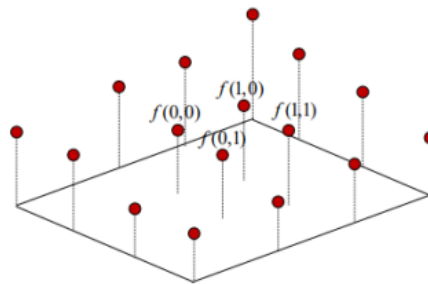
Diberikan sebuah matrix awal, misal  $M$ , kita akan mencari persamaan interpolasi  $f(x, y)$  dengan pemodelan sebagai berikut:

**Normalization:**  $f(0,0), f(1,0)$

$f(0,1), f(1,1)$

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

**Solve:**  $a_{ij}$

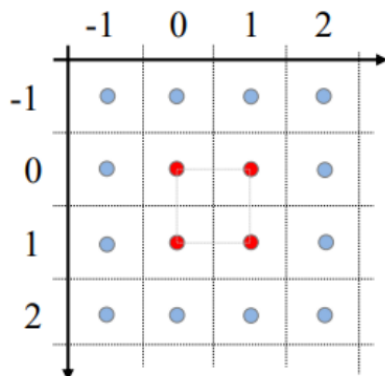


Melakukan substitusi nilai-nilai diketahui pada matriks 4 x 4 tersebut ke persamaan  $f(x,y)$  akan menghasilkan sebuah matriks persamaan:

$$y = Xa$$

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

Elemen pada matrix X adalah koefisien  $a_{ij}$  yang diperoleh dari persamaan  $f(x,y)$  di atas. Sebagai contoh, elemen pada baris 4 kolom ke 10 adalah koefisien dari  $a_{12}$  dan diperoleh dari  $2^1 \cdot (-1)^2 = 2$ , sesuai persamaan  $x^i \cdot y^j$ . 3 Vektor  $a$  dapat dicari dari persamaan tersebut (menggunakan inverse), lalu vektor  $a$  digunakan sebagai nilai variabel dalam  $f(x,y)$ . Sehingga terbentuk fungsi interpolasi bicubic sesuai model. Tugas Anda adalah menentukan persamaan  $f(x,y)$  lalu melakukan interpolasi berdasarkan  $f(a,b)$  dari masukan matriks 4 x 4. Nilai masukan  $a$  dan  $b$  dalam rentang  $[0,1]$ .



(nilai untuk diinterpolasi dalam kotak merah)

## X. Regresi Linier Berganda

Regresi Linier Berganda merupakan salah satu metode untuk memprediksi nilai selain menggunakan Interpolasi Polinom. Terdapat rumus umum dari regresi linier yang bisa digunakan untuk regresi linier berganda, yaitu

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

Untuk mendapat setiap nilai  $\beta_i$  dapat digunakan *Normal Estimation Equation for Multiple Linear Regression* sebagai berikut

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

Kemudian sistem persamaan linier tersebut diselesaikan dengan metode Gauss.

## BAB 3

### Implementasi Pustaka dan Program

#### I. Class Main

Nama	Tipe	Parameter	Deskripsi
main	Public static void	String[] args	Memulai program utama
displayCommand	Public static void	-	Menampilkan pesan untuk memasukkan input
displayMenu	Public static void	-	Menampilkan menu utama
displaySubSPL	Public static void	-	Menampilkan <i>sub-menu</i> SPL
displaySubDet	Public static void	-	Menampilkan <i>sub-menu</i> determinan
displaySubInverse	Public static void	-	Menampilkan <i>sub-menu</i> invers matriks
displaySubInterpol	Public static void	-	Menampilkan <i>sub-menu</i> Interpolasi Polinom
displaySubBicubic	Public static void	-	Menampilkan <i>sub-menu</i> Interpolasi Bikubik
displaySubRegression	Public static void	-	Menampilkan <i>sub-menu</i> Regresi Linier Berganda
displayInputType	Public static void	-	Menampilkan pilihan input program
displaySavePromptGauss	Public static void	MATRIKS Matrix, MATRIKS hasil	Menyimpan hasil perhitungan Gauss ke dalam file
displaySavePromptBicubic	Public static void	MATRIKS Matrix, double[] xy, double hasil	Menyimpan hasil perhitungan Interpolasi Bicubic ke

			dalam file
optionBranch	Public static void	Int option	Memulai program sesuai pilihan dari menu utama
SPL	Public static void	-	Memulai perhitungan SPL sesuai metode yang dipilih
gaussianElim	Public static void	-	Memulai perhitungan SPL dengan metode Eliminasi Gauss dengan input sesuai yang user inginkan
gaussjordanElim	Public static void	-	Memulai perhitungan SPL dengan metode Eliminasi Gauss-Jordan dengan input sesuai yang user inginkan
inverseEq	Public static void	-	Memulai perhitungan SPL dengan metode invers matriks dengan input sesuai yang user inginkan
cramer	Public static void	-	Memulai perhitungan SPL dengan metode Cramer dengan input sesuai yang user inginkan
determinant	Public static void	-	Memberikan pilihan kepada user untuk metode mencari determinan matriks dan memulainya sesuai pilihan
detMetode1	Public static void	-	Memulai perhitungan determinan dengan metode Ekspansi Kofaktor dengan input sesuai yang user inginkan
detMetode2	Public static void	-	Memulai perhitungan determinan dengan

			metode Reduksi Baris (Gauss) dengan input sesuai yang user inginkan
inverse	Public static void	-	Memberikan pilihan kepada user untuk metode mencari invers matriks dan memulainya sesuai pilihan
invMetode1	Public static void	-	Memulai perhitungan invers matriks dengan metode Eliminasi Gauss-Jordan dengan input sesuai yang user inginkan
invMetode2	Public static void	-	Memulai perhitungan invers matriks dengan metode Kofaktor dengan input sesuai yang user inginkan
polInter	Public static void	-	Memberikan pilihan kepada user untuk metode mencari Interpolasi Polinom dan memulainya sesuai pilihan
polMetode1	Public static void	-	Memulai perhitungan Interpolasi Polinom dengan input sesuai yang user inginkan
cubInter	Public static void	-	Memberikan pilihan kepada user untuk metode mencari Interpolasi Bikubik dan memulainya sesuai pilihan
bicInterpolation	Public static void	-	Memulai perhitungan Interpolasi Bikubik dengan input sesuai yang user inginkan
regression	Public static void	-	Memberikan pilihan kepada user untuk



			metode mencari Regresi Linier Berganda dan memulainya sesuai pilihan
regMetode1	Public static void	-	Memulai perhitungan Regresi Linier Berganda dengan input sesuai yang user inginkan

## II. Class InvMatrix

Nama	Tipe	Parameter	Deskripsi
InvStartMethod1	Public Static void	Boolean keyboardInput	Menghasilkan invers matriks dengan metode Eliminasi Gauss-Jordan
InvStartMethod2	Public Static void	Boolean keyboardInput	Menghasilkan invers matriks dengan metode Kofaktor
ReadMatriksSym	Public Static double[][]	-	Mengambil input matriks persegi/simetri dari user
InvGaussJordan	Public Static double[][]	Double[][] Mat	Mengembalikan hasil invers matriks dengan metode Eliminasi Gauss-Jordan
InvCofactor	Public static double[][]	Double[][] Mat	Mengembalikan hasil invers matriks dengan metode Kofaktor
displaySavePromptInverse	Public static void	Double[][] InvMat	Menampilkan penawaran untuk menyimpan hasil

			dalam file atau tidak dan mengeksekusinya
writeFileInverse	Public static void	String FileName, double[][] InvMat	Menuliskan hasil dari program ke sebuah file baru

### III. Class InterPol

Nama	Tipe	Parameter	Deskripsi
InterPolaStart1	Public static void	-	Menghasilkan fungsi interpolasi dengan input dari keyboard
InterPolaStart2	Public static void	-	Menghasilkan fungsi interpolasi dengan input dari file
SolvedFuncInterPol1	Public static Double[]	Double[][] X, int n	Mengeksekusi proses perhitungan interpolasi dengan input dari keyboard
SolvedFuncInterPol2	Public static Double[]	-	Mengeksekusi proses perhitungan interpolasi dengan input dari file
EstimateInterPol	Public static Double	Double[] Result, double X	Menampilkan estimasi hasil dari fungsi interpolasi yang diperoleh
ReadPoint	Public static Void	Double[] X, int n	Membaca input user dan membentuk matriks
CreateMatInterPol	Public static double[][]	Double[][] X	Membuat matriks dari titik-titik interpolasi yang didapat
SPLResult	Public static double[]	Double[][] Mat	Mengambil hasil SPL dari matriks interpolasi
printFunct	Public static void	Double[] Res	Menuliskan hasil interpolasi dalam bentuk fungsi ke layar

displaySavePromptPol Inter	Public static void	Double[] Result, double Estimated, double X	Menampilkan penawaran untuk menyimpan hasil dalam file atau tidak dan mengeksekusinya
writeFilePolInter	Public static void	String FileName, double[] Result, double Estimated, double X	Menuliskan hasil program ke sebuah file baru

#### IV. Class Gauss

Nama	Tipe	Parameter	Deskripsi
GaussSolver	Public static String[]	MATRIKS M	Mengembalikan solusi dari matriks berbentuk string gabungan nilai dan parameter
checkResult	Public static void	MATRIKS M, int i, String[] resultString	Mengembalikan string solusi eliminasi gauss
isElmtZero	public static boolean	MATRIKS M, int i, int j	Mengecek jika elemen matriks bernilai 0
isAllInRowZero	public static boolean	MATRIKS M, int i	Mengembalikan nilai true jika semua nilai dalam baris bernilai nol
isSolZero	public static boolean	MATRIKS M, int i	Mengembalikan nilai true jika pada kolom terakhir terdapat nilai 0 pada sebuah baris indeks i
hasSolution	public static boolean	MATRIKS M	Menghasilkan True jika Matriks mempunyai solusi dan False jika tidak
addParameter	private static String	double koefisien, String parameter	Menggabungkan hasil string dan parameter
isOneLeadElmt	public static boolean	int i, MATRIKS M	Mengembalikan nilai true jika Matriks M

			baris ke-i hanya memiliki satu leadElement
leadElmtIndex	public static int	int i, MATRIKS M	Mengembalikan Posisi dari leadingElement baris ke-i
reducedRE	Public static void	MATRIKS M	Membentuk matriks eselon baris tereduksi dari matriks M
rowEchelon	Public static void	MATRIKS M	Mengubah matriks M menjadi bentuk Row Echelon
upperTriangle	public static void	MATRIKS M	Membentuk matriks segitiga atas dari Matriks M
findRowZero	public static void	MATRIKS M, int k	Mencari jumlah nol di baris ke-i dan semua baris setelahnya
printSolution	public static String[]	MATRIKS M	Mengeluarkan output solusi dari SPL
defineSolution	public static String[]	MATRIKS M, String[] solution, String[] Char	Menentukan jenis solusi yang ditemukan dan menentukan format untuk mencetak solusi
writeFileGauss	public static void	String FileName, MATRIKS Matrix, MATRIKS hasil	Menulis hasil perhitungan gauss ke file

## V. Class MATRIKS

Nama	Tipe	Parameter	Deskripsi
MATRIKS	Konstruktor	int baris, int kolom	Konstruktor dengan input

MATRIKS	Konstruktor	String path	Konstruktor metode file
getRow	public int	MATRIKS M, int i, int j	Mengembalikan nilai ukuran baris dari sebuah matriks
getCol	public int	MATRIKS M, int i	Mengembalikan nilai ukuran kolom dari sebuah matriks
setRow	public void	int N	Membentuk matriks dengan baris sebanyak N
setCol	public void	int N	Membentuk matriks dengan kolom sebanyak N
getElmt	public static double	MATRIKS M, int baris, int kolom	Mengembalikan elemen sebuah matriks
readMatrix	public static void	MATRIKS Matrix	Membaca matriks dari input user
readFileMatrix	public void	String path	Membaca matriks dari file
printMatrix	Public static void	MATRIKS Matrix	Menampilkan matriks pada layar
printMatriks	Public static void	double[][] Mat	Menampilkan matriks pada layar
Cofactor	public static double[][]	double[][] Mat	Mengembalikan matriks kofaktor dari sebuah matriks
isColZero	public static boolean	double[][] mat, int inCol	Menentukan apakah kolom tertentu berelemen semua nol atau tidak
minor	public static double[][]	double[][] mat, int Row, int Col	Menghasilkan matriks minor dari elemen pada sebuah matriks
transpose	public static double[][]	double[][] mat	Menghasilkan hasil transpose sebuah

			matriks
multiplyMatbyConst	public static double[][]	double[][] mat, double x	Menghasilkan matriks yang telah dikalikan dengan sebuah konstanta
CopyMatrix	public static double[][]	double[][] mat	Menghasilkan matriks yang serupa dengan matriks parameter
copyMatriks	public static MATRIKS	MATRIKS M	Menghasilkan matriks yang identik dengan matriks parameter
reducedRE	public static void	double[][] M	Melakukan perhitungan dengan metode Gauss-jordan pada matriks yang sudah berbentuk matriks Echelon baris
rowEchelon	public static void	double[][] M	Melakukan perhitungan dengan metode Gauss pada matriks
leadElmtIndex	public static int	int i, double[][] M	Menentukan lead elmt dari matriks
readMatFile	public static double[][]	String FileName	Membaca matriks dari file

## VI. Class bicubicInt

Nama	Tipe	Parameter	Deskripsi
bicubicInterpolation	public static void	MATRIKS M	Menerima input nilai f serta x dan y lalu mengembalikan nilai interpolasi bikubik
inputPoints	public static void	MATRIKS M	Memberikan prompt pada user untuk

			mengisi nilai $f(-1,-1)$ sampai $f(2,2)$
inputXY	public static double[]	-	Memberikan prompt pada user untuk mengisi nilai x dan y yang ditaksir
findCubicInter	public static double	MATRIKS M	Mengembalikan nilai interpolasi bikubik dari nilai f yang dicari
writeFileBicubic	public static void	String FileName, MATRIKS Matrix, double[]xy, double hasil	Menulis hasil perhitungan interpolasi bikubik ke file

## VII. Class Regarr

Nama	Tipe	Parameter	Deskripsi
Regarr	Konstruktor	int Len	Membuat sebuah Array dengan panjang Len
getLen	Public int	-	Mengirim panjang Array
setLen	Public void	int Len	Mengubah panjang Array menjadi Len
getArrElmt	Public double	Regarr array, int Col	Mengirim elemen indeks Array ke-Col
readArr	Public static void	Regarr Array, int Len	Membuat Array dengan panjang Len, menerima input elemen Array
Convert	Public static void	Regarr numArr, String[] strArr	Mengubah tipe data elemen array dari string ke double
printArr	Public static void	Regarr Array	Menampilkan Array
printFunct	Public static void	Regarr Array	Menampilkan Array sesuai ketentuan (dalam bentuk fungsi)

## VIII. Class Regmat

Nama	Tipe	Parameter	Deskripsi
Regmat	Konstruktor	int Row, int Col	Membuat sebuah Matriks dengan panjang baris Row dan panjang kolom Col
getRow	Public int	-	Mengirim panjang baris matriks
getCol	Public int	-	Mengirim panjang kolom matriks
setRow	Public void	Int Row	Mengubah panjang baris matriks menjadi Row
setCol	Public void	Int Col	Mengubah panjang kolom menjadi Col
getElmt	Public double	Regmat Matrix, int Row, int Col	Mengirim elemen matrix pada indeks baris ke-Row dan kolom ke-Col
readMat	Public static void	Regmat Matrix, int Row, int Col	Membuat sebuah matriks dengan panjang baris row dan panjang kolom Col, menerima input elemen matriks
readFile	Public static void	Regmat Matrix, int Row, int Col	Membaca file dan membuat matriks sesuai dengan isi file
makeRegMat	Public static void	Regmat regMat, int varCount, int sampleCount, regMat inputMat	Menerima jumlah peubah, jumlah sampel, dan matriks input dan membuat matriks Normal Estimation Equation for Multiple Linear Regression
printMat	Public static void	Regmat Matrix	Menampilkan matriks



## IX. Class Regression

Nama	Tipe	Parameter	Deskripsi
GaussSolver	Public static String[]	Regmat M	Mengembalikan solusi dari matriks berbentuk string gabungan nilai dan parameter
checkResult	Public static void	Regmat M, int i, String[] resultString	Mengembalikan string solusi eliminasi gauss
isElmtZero	public static boolean	Regmat M, int i, int j	Mengecek jika elemen matriks bernilai 0
isAllInRowZero	public static boolean	Regmat M, int i	Mengembalikan nilai true jika semua nilai dalam baris bernilai nol
isSolZero	public static boolean	Regmat M, int i	Mengembalikan nilai true jika pada kolom terakhir terdapat nilai 0 pada sebuah baris indeks i
hasSolution	public static boolean	Regmat M	Menghasilkan True jika Matriks mempunyai solusi dan False jika tidak
addParameter	private static String	Double koefisien, String parameter	Menggabungkan hasil string dan parameter
isOneLeadElmt	public static boolean	Int i, Regmat M	Mengembalikan nilai true jika Matriks M baris ke-i hanya memiliki satu leadElement
leadElmtIndex	public static int	Int i, Regmat M	Mengembalikan Posisi dari leadingElement baris ke-i
rowEchelon	Public static void	Regmat M	Mengubah matriks M menjadi bentuk Row Echelon
printSolution	public static String[]	Regmat M	Mengeluarkan output solusi dari SPL

defineSolution	public static String[]	Regmat M, Solution[] solution, String[] char	Menentukan jenis solusi yang ditemukan dan menentukan format untuk mencetak solusi
savePromt	Public static void	Regarr ResArr, double RegRes, RegArr inputArr	Menanyakan user apakah ingin meyimpan hasil kalkulasi regresi
writeFile	Public static void	String FileName, Regarr ResArr, double RegRes, Regarr inputArr	Menuliskan dan meyimpan hasil kalkulasi regresi
RegCalc1	Public static void	-	Menerima input n, m, nilai xin, yi, xk, dan melakukan kalkulasi persamaan regresi dan hasil taksir xk
RegCalc2	Public static void	-	Menerima input nama file, membuat matrix sesuai isi file, menerima input xk, melakukan kalkulasi persamaan regresi dan hasil taksir xk

#### X. Class Detmat

Nama	Tipe	Parameter	Deskripsi
Detmat	Konstruktor	Int Row, int Col	Membuat matriks dengan panjang baris Row dan panjang kolom Col
getRow	Public int	-	Mengirim panjang baris matriks
getCol	Public int	-	Mengirim panjang kolom matriks
setRow	Public void	Int Row	Mengubah panjang baris matriks menjadi Row

setCol	Public void	Int Col	Mengubah panjang kolom menjadi Col
getElmt	Public double	Detmat Matrix, int Row, int Col	Mengirim elemen matrix pada indeks baris ke-Row dan kolom ke-Col
readMat	Public static void	Detmat Matrix, int Row, int Col	Membuat sebuah matriks dengan panjang baris row dan panjang kolom Col, menerima input elemen matriks
readFile	Public static void	String fileName, Detmat Matrix	Membaca file dan membuat matriks sesuai dengan isi file
printMat	Public static void	Detmat Matrix	Menampilkan matriks

## XI. Class Reduction

Nama	Tipe	Parameter	Deskripsi
swapRow	Public static void	Detmat Matrix, int curRow, int targetRow	Menerima matriks dan posisi baris curRow dengan targetRow
makeOne	Public static void	Detmat Matrix, int curRow, int curCol	Menerima matriks dan membuat leading elemen curRow pada kolom curCol menjadi 1
makeZero	Public static void	Detmat Matrix, int curRow, int prevRow	Menerima matriks dan membuat elemen curRow menjadi 0 berdasarkan elemen prevRow
leadVal	Public static double	Detmat Matrix, int curRow, int curCol	Mengirim leading elemen matriks
RedSolve	Public static double	Detmat Matrix	Menerima matriks dan menghitung determinan matriks dengan OBE
savePromt	Public static void	Double redRes	Menanyakan apakah

			user ingin menyimpan hasil determinan
writeFile	Public static void	String fileName, double redRes	Menuliskan dan menyimpan hasil determinan ke file .txt
RedCalc1	Public static void	-	Menerima input n dan koefisien aij, dan melakukan kalkulasi determinan matriks
RedCalc2	Public static void	-	Menerima input nama file, melakukan kalkulasi determinan matriks

## XII. Class Expansion

Nama	Tipe	Parameter	Deskripsi
Minor	Public static Detmat	Detmat Matrix, int curCol	Menerima matriks dan mengembalikan matriks minornya
Recursion	Public static double	Detmat Matrix, int curCol	Menerima matriks dan melakukan rekursi sampai ditemukan minor 2x2 lalu mengembalikan determinannya
Cofactor	Public static int	Int curCol	Mengembalikan positif atau negatif berdasarkan posisi kolom elemen matriks
DetMinor	Public static double	Detmat Matrix	Mengembalikan determinan matriks berukuran 2x2
savePromt	Public static void	Double redRes	Menanyakan apakah user ingin menyimpan hasil determinan
writeFile	Public static void	String fileName, double redRes	Menuliskan dan menyimpan hasil determinan ke file .txt
ExpCalc2	Public static void	-	Menerima input nama file, melakukan kalkulasi

			determinan matriks
--	--	--	--------------------

### XIII. Class Cramarr

Nama	Tipe	Parameter	Deskripsi
Cramarr	Konstruktor	Int Len	Membuat Array dengan panjang Len
getLen	Public int	-	Mengirim panjang Array
setLen	Public void	int Len	Mengubah panjang Array menjadi Len
getArrElmt	Public double	Cramarr array, int Col	Mengirim elemen indeks Array ke-Col
printArr	Public static void	Cramarr Array	Menampilkan Array

### XIV. Class Cramat

Nama	Tipe	Parameter	Deskripsi
Cramat	Konstruktor	Int Row, int Col	Membuat Matrix dengan panjang baris Row dan panjang kolom Col
getRow	Public int	-	Mengirim panjang baris matriks
getCol	Public int	-	Mengirim panjang kolom matriks
setRow	Public void	Int Row	Mengubah panjang baris matriks menjadi Row
setCol	Public void	Int Col	Mengubah panjang kolom menjadi Col
getElmt	Public double	Cramat Matrix, int Row, int Col	Mengirim elemen matrix pada indeks baris ke-Row dan kolom ke-Col
readMat	Public static void	Cramat Matrix, int Row, int Col	Membuat sebuah matriks dengan panjang baris row dan panjang kolom Col, menerima input elemen matriks

readFile	Public static void	String fileName, Cramat Matrix	Membaca file dan membuat matriks sesuai dengan isi file
mainMat	Public static void	Cramat inpMat, Cramat outMat	Membuat matriks utama outMat (tanpa kolom bi) berdasarkan matriks augmented inpMat
curMat	Public static void	Cramat inpMat, Cramat outMat, int curCol	Membuat salinan matriks utama dan mengubah elemen pada indeks kolom ke-curCol dengan bi, elemen kolom terakhir pada matriks augmented inpMat
swapRow	Public static void	Cramat Matrix, int curRow, int targetRow	Menerima matriks dan posisi baris curRow dengan targetRow
makeOne	Public static void	Cramat Matrix, int curRow, int curCol	Menerima matriks dan membuat leading elemen curRow pada kolom curCol menjadi 1
makeZero	Public static void	Cramat Matrix, int curRow, int prevRow	Menerima matriks dan membuat elemen curRow menjadi 0 berdasarkan elemen prevRow
leadVal	Public static double	Cramat Matrix, int curRow, int curCol	Mengirim leading elemen matriks
findDet	Public static double	Cramat Matrix	Menerima matriks dan menghitung determinan matriks dengan OBE
printMat	Public static void	Cramat Matrix	Menampilkan matriks

#### XV. Class Cramer

Nama	Tipe	Parameter	Deskripsi
savePromt	Public static void	Double mainDet, Cramarr solArr	Menanyakan apakah user ingin menyimpan hasil kalkulasi solusi SPL dengan metode cramer

writeFile	Public static void	String FileName, double mainDet, Cramarr solArr	Menuliskan dan menyimpan hasil kalkulasi solusi SPL dengan metode cramer
CramCalc1	Public static void	-	Menerima masukan n, masukan elemen matriks, melakukan kalkulasi solusi SPL dengan metode cramer
CramCalc2	Public static void	-	Menerima masukan nama file, membaca file, membuat matriks sesuai isi file, melakukan kalkulasi solusi SPL dengan metode cramer

#### XVI. Class InvSpl

Nama	Tipe	Parameter	Deskripsi
readMatFile	Public static double[][]	String FileName	Membaca isi file dan membuat matrix sesuai dengan isi file
findSol	Public static void	Boolean keyboardInput	Menerima boolean keyboardInput dan menentukan apakah matriks dibuat berdasarkan masukan user atau file, kemudian melakukan kalkulasi solusi SPL dengan metode inverse matriks
ReadMatriks	Public static double[][]	-	Membaca input n dan koefisien aij matriks
InvGaussJordan	Public static double[][]	Double[][] mat	Melakukan invers terhadap mat dengan metode gauss jordan
displaySavePrompt	Public static void	Double det, double[] solArr	Menanyakan apakah user ingin menyimpan hasil kalkulasi solusi SPL dengan metode inverse matriks
writeFileInverse	Public static void	String FileName,	Menuliskan dan

		double det, double[] solArr	menyimpan hasil kalkulasi solusi SPL dengan metode cramer
--	--	--------------------------------	---

## XVII. Class OBE

Nama	Tipe	Parameter	Deskripsi
printMatrix	Static void	Double[][] dataArray	Menampilkan matrix
swapRow	Public static void	Double[][] dataArray, int curRow, int targetRow	Menerima matriks dan posisi baris curRow dengan targetRow
makeOne	Public static void	Double[][] dataArray, int curRow, int curCol	Menerima matriks dan membuat leading elemen curRow pada kolom curCol menjadi 1
makeZero	Public static void	Double[][] dataArray, int curRow, int prevRow	Menerima matriks dan membuat elemen curRow menjadi 0 berdasarkan elemen prevRow
leadVal	Public static double	Double[][] dataArray, int curRow, int curCol	Mengirim leading elemen matriks
RedSolve	Public static double	Double[][] dataArray	Menerima matriks dan menghitung determinan matriks dengan OBE



## BAB 4

### Eksperimen

#### I. Solusi Sistem Persamaan Linier berbentuk $Ax = b$

1. Temukan solusi SPL  $Ax = b$ , berikut:

a.

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

b.

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

c.

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

d.

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

Solusi a:

```
Matriks hasil operasi :  
1.00 0.00 0.00 0.67 0.00  
0.00 1.00 0.00 -2.67 0.00  
0.00 0.00 1.00 -1.00 0.00  
0.00 0.00 0.00 0.00 1.00  
  
There is no solution
```

Matriks pertama tidak memiliki solusi karena koefisien variabel semua bernilai 0 tapi jumlah bernilai tidak nol

Solusi b:

```
Input Matrix :  
  
1.00 -1.00 0.00 0.00 1.00 3.00  
1.00 1.00 0.00 -3.00 0.00 6.00  
2.00 -1.00 0.00 1.00 -1.00 5.00  
-1.00 2.00 0.00 -2.00 -1.00 -1.00  
  
Matriks hasil operasi :  
  
1.00 0.00 0.00 0.00 -1.00 3.00  
0.00 1.00 0.00 0.00 -2.00 0.00  
0.00 0.00 0.00 1.00 -1.00 -1.00  
0.00 0.00 0.00 0.00 0.00 0.00  
  
Solusi dari matrix adalah :  
x1 = 3.0+d  
x2 = 0.0+2.0d  
x3 = b  
x4 = -1.0+d  
x5 = d
```

Matriks kedua memiliki solusi banyak dengan bentuk parametrik seperti tertera karena koefisien variabel dan hasil bernilai 0

Solusi c:

```
Input Matrix :  
  
0.00 1.00 0.00 0.00 1.00 0.00 2.00  
0.00 0.00 0.00 1.00 1.00 0.00 -1.00  
0.00 1.00 0.00 0.00 0.00 1.00 1.00  
  
Matriks hasil operasi :  
  
0.00 1.00 0.00 0.00 0.00 1.00 1.00  
0.00 0.00 0.00 1.00 0.00 1.00 -2.00  
0.00 0.00 0.00 0.00 1.00 -1.00 1.00  
  
Solusi dari matrix adalah :  
x1 = ~  
x2 = 1.0-e  
x3 = b  
x4 = -2.0-e  
x5 = 1.0+e  
x6 = e
```

Matriks ketiga memiliki solusi banyak dengan bentuk parametrik seperti tertera karena koefisien variabel dan hasil bernilai 0

Solusi d:

```
Input Matrix :  
1.00 0.50 0.33 0.25 1.00  
0.50 0.33 0.25 0.20 0.00  
0.33 0.25 0.20 0.17 0.00  
0.25 0.20 0.17 0.14 0.00  
  
Matriks hasil operasi :  
1.00 0.50 0.33 0.25 1.00  
0.00 1.00 1.06 0.94 -6.25  
0.00 0.00 1.00 4.84 255.56  
0.00 0.00 0.00 1.00 81.51  
  
Solusi dari matrix adalah :  
x1 = -6.027397260274039  
x2 = 65.06849315068557  
x3 = -139.04109589041238  
x4 = 81.50684931506937
```

Matriks keempat memiliki solusi unik seperti tertera diatas

II. Solusi Sistem Persamaan Linier berbentuk matriks *Augmented*

2. SPL berbentuk matriks *augmented*

a.

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

b.

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

Solusi a:

```
Matriks hasil operasi :  
1.00 0.00 0.00 -1.00 -1.00  
0.00 1.00 -2.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00
```

```
Solusi dari matrix adalah :  
x1 = -1.0+c  
x2 = 0.0+2.0b  
x3 = b  
x4 = c
```

Matriks pertama memiliki solusi banyak dengan bentuk parametrik seperti tertera karena koefisien variabel dan hasil bernilai 0

Solusi b:

```
Matrks hasil operasi :  
1.00 0.00 4.00 0.00 4.00  
0.00 1.00 0.00 4.00 6.00  
0.00 0.00 1.00 0.00 1.00  
0.00 0.00 0.00 1.00 1.00  
0.00 0.00 0.00 0.00 0.00  
0.00 0.00 0.00 0.00 0.00  
  
Solusi dari matrix adalah :  
x1 = 0.0  
x2 = 2.0  
x3 = 1.0  
x4 = 1.0
```

Matriks kedua memiliki solusi unik seperti tertera diatas

### III. Solusi Sistem Persamaan Linier berbentuk persamaan

#### 3. SPL berbentuk

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

b.

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

Solusi a:

```
Matriks hasil operasi :  
1.00 0.00 0.00 0.00 -0.22  
0.00 1.00 0.00 0.00 0.18  
0.00 0.00 1.00 0.00 0.71  
0.00 0.00 0.00 1.00 -0.26
```

```
Solusi dari matrix adalah :  
x1 = -0.22432432432432425  
x2 = 0.18243243243243235  
x3 = 0.7094594594594595  
x4 = -0.25810810810810836
```

SPL pertama memiliki solusi unik dengan solusi seperti tertera

Solusi b:

```
Input Matrix :  
0.00 0.00 0.00 0.00 0.00 0.00 1.00 1.00 1.00 13.00  
0.00 0.00 0.00 1.00 1.00 1.00 0.00 0.00 0.00 15.00  
1.00 1.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00 8.00  
0.00 0.00 0.04 0.00 0.04 0.75 0.04 0.75 0.61 14.79  
0.00 0.25 0.91 0.25 0.91 0.25 0.91 0.25 0.00 14.31  
0.61 0.75 0.04 0.75 0.04 0.00 0.04 0.00 0.00 3.81  
0.00 0.00 1.00 0.00 0.00 1.00 0.00 0.00 1.00 18.00  
0.00 1.00 0.00 0.00 1.00 0.00 0.00 1.00 0.00 12.00  
1.00 0.00 0.00 1.00 0.00 0.00 1.00 0.00 0.00 6.00  
0.04 0.75 0.61 0.00 0.04 0.75 0.00 0.00 0.04 10.51  
0.91 0.25 0.00 25.00 0.91 25.00 0.00 25.00 0.91 16.13  
0.04 0.00 0.00 0.75 0.04 0.00 0.61 0.75 0.04 7.04
```

```
Matriks hasil operasi :  
1.00 1.00 1.00 0.00 0.00 0.00 0.00 0.00 0.00 8.00  
1.00 1.22 0.07 1.22 0.07 0.00 0.00 -0.07 -0.07 5.30  
0.00 1.00 0.06 0.18 0.06 0.00 0.00 0.76 0.76 10.07  
0.00 0.00 1.00 -0.31 0.00 1.31 0.00 -0.92 -0.85 5.71  
0.00 0.00 0.00 1.00 0.04 1.04 0.00 1.07 0.07 0.84  
0.00 0.00 0.00 0.00 1.00 0.30 0.00 0.42 -0.84 2.55  
0.00 0.00 0.00 0.00 0.00 1.00 0.00 0.54 -0.22 4.64  
0.00 0.00 0.00 0.00 0.00 0.00 1.00 1.00 1.00 13.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00 1.80 16.03  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00 -13.33  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 1.00  
0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
```

There is no solution

SPL kedua tidak memiliki solusi karena koefisien variabel semua bernilai 0 tapi jumlah bernilai tidak nol

#### IV. Studi kasus Interpolasi Polinom

##### a. Kasus pertama

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

$x$	0.4	0.7	0.11	0.14	0.17	0.2	0.23
$f(x)$	0.043	0.005	0.058	0.072	0.1	0.13	0.147

Lakukan pengujian pada nilai-nilai default berikut:

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

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

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

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

Persamaan polinom yang diperoleh adalah

$$f(x) = -0.1846 + 10.2764x - 163.9157x^2 + 1220.8549x^3 - 4346.3140x^4 + 7102.3992x^5 - 4212.4345x^6$$

$$f(x) = -0.1846 + 10.2764x - 163.9157x^2 + 1220.8549x^3 - 4346.3140x^4 + 7102.3992x^5 - 4212.4345x^6$$

(setiap koefisien fungsi ditulis dalam format empat angka di belakang koma)

$$f(0.2) = 0.1300$$

$$\text{Hasil taksiran fungsi } f(0.2) = 0.1300$$

( $x$  dalam  $f(x)$  ditulis dalam format satu angka di belakang koma)

$$f(0.55) = 2.1376$$

$$\text{Hasil taksiran fungsi } f(0.6) = 2.1376$$

( $x$  dalam  $f(x)$  ditulis dalam format satu angka di belakang koma)

$$f(0.85) = -66.2696$$

$$\text{Hasil taksiran fungsi } f(0.850) = -66.2696$$

( $x$  dalam  $f(x)$  ditulis dalam format tiga angka di belakang koma)

$$f(1.28) = -3485.1449$$

$$\text{Hasil taksiran fungsi } f(1.280) = -3485.1449$$

( $x$  dalam  $f(x)$  ditulis dalam format tiga angka di belakang koma)

b. Kasus kedua

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

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

Tanggal (desimal) adalah tanggal yang sudah diolah ke dalam bentuk desimal 3 angka di belakang koma dengan memanfaatkan perhitungan sebagai berikut:

$$\text{tanggal(desimal)} = \text{bulan} + (\text{tanggal} / \text{jumlah hari pada bulan tersebut})$$

Sebagai **contoh**, untuk tanggal 17/06/2022 (dibaca: 17 Juni 2022) diperoleh tanggal(desimal) sebagai berikut:

$$\text{Tanggal(desimal)} = 6 + (17/30) = 6,567$$

Gunakanlah data di atas dengan memanfaatkan **polinom interpolasi** untuk melakukan prediksi jumlah kasus baru Covid-19 pada tanggal-tanggal berikut:

- 16/07/2022
- 10/08/2022
- 05/09/2022
- beserta masukan user lainnya berupa **tanggal (desimal) yang sudah diolah** dengan asumsi prediksi selalu dilakukan untuk tahun 2022.



Input file

```
test > ≡ 4B.txt
1 6.657 12.624
2 7 21.807
3 7.258 38.391
4 7.451 54.517
5 7.548 51.952
6 7.839 28.228
7 8.161 35.764
8 8.484 20.813
9 8.709 12.408
10 9 10.534
```

Persamaan polinom yang diperoleh adalah

```
Masukkan nama file :
4B.txt

f(x) = 2344115157.2549 + -3820124716.9969X + 2533237389.6482X^2 + -929450771.3937X^3 +
211574187.3831X^4 + -31292717.1359X^5 + 3025646.2133X^6 + -185162.7891X^7 + 6526.5328X^8 + -101.1578X^9
```

$$f(x) = 2344115157.2549 + -3820124716.9969x + 2533237389.6482x^2 + -929450771.3937x^3 + 211574187.3831x^4 + -31292717.1359x^5 + 3025646.2133x^6 + -185162.7891x^7 + 6526.5328x^8 + -101.1578x^9$$

16/07/2022

$$x = 7 + (16/31) = 7.516$$

Hasil taksiran fungsi  $f(7.516) = 53.5417$

10/08/2022

$$x = 8 + (10/31) = 8.323$$

Hasil taksiran fungsi  $f(8.323) = 36.1698$

05/09/2022

$$x = 9 + (5/30) = 9.167$$

Hasil taksiran fungsi  $f(9.167) = -635.7627$

21/11/2022

$$x = 11 + (21/30) = 11.7$$

Hasil taksiran fungsi  $f(11.700) = -40397908.8884$

c. Kasus ketiga

Sederhanakan fungsi

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

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

Input file

```
test > ≡ 4C.txt
1    0 0
2    0.4 0.418884
3    0.8 0.507158
4    1.2 0.560925
5    1.6 0.583686
6    2 0.576651
```

Fungsi yang diperoleh adalah

```
Masukkan nama file :
4C.txt

f(x) = 0.0000 + 2.0353X + -3.5527X^2 + 3.2371X^3 + -1.4213X^4 + 0.2363X^5
```

$$f(x) = 0.0000 + 2.0353x + -3.5527x^2 + 3.2371x^3 + -1.4213x^4 + 0.2363x^5$$

## V. Studi kasus Interpolasi Bikubik

### 4. Studi Kasus Interpolasi Bicubic

Diberikan matriks input:

153	59	210	96
125	161	72	81
98	101	42	12
21	51	0	16

Tentukan nilai:

$$f(0,0) = ?$$

$$f(0.5, 0.5) = ?$$

$$f(0.25, 0.75) = ?$$

$$f(0.1, 0.9) = ?$$

Berdasarkan hasil perhitungan program interpolasi bikubik kami nilai-nilai diatas dapat ditentukan sebagai berikut:

File Matriks :

```
153 59 210 96
125 161 72 81
98 101 42 12
21 51 0 16
```

1. Hasil dari  $f(0,0)$  :

```
Mencari nilai f(x,y) dengan interpolasi bikubik

Input nilai x :
0
Input nilai y :
0

Nilai dari f(0.000,0.000) adalah 161.000
```

2. Hasil dari  $f(0.5,0.5)$  :

```
Mencari nilai  $f(x,y)$  dengan interpolasi bikubik  
Input nilai x :  
0.5  
Input nilai y :  
0.5  
  
Nilai dari  $f(0.500,0.500)$  adalah 97.727
```

3. Hasil dari  $f(0.25,0.75)$  :

```
Mencari nilai  $f(x,y)$  dengan interpolasi bikubik  
Input nilai x :  
0.25  
Input nilai y :  
0.75  
  
Nilai dari  $f(0.250,0.750)$  adalah 105.515
```

4. Hasil dari  $f(0.1,0.9)$  :

```
Mencari nilai  $f(x,y)$  dengan interpolasi bikubik  
Input nilai x :  
0.1  
Input nilai y :  
0.9  
  
Nilai dari  $f(0.100,0.900)$  adalah 104.229
```

## VI. Studi kasus Regresi Linier Berganda

Table 12.1: Data for Example 12.1

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

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

Input file:

test > 5.txt

You, 22 hours ago | 1 author (You)

1	72.4	76.3	29.18	0.90
2	41.6	70.3	29.35	0.91
3	34.3	77.1	29.24	0.96
4	35.1	68.0	29.27	0.89
5	10.7	79.0	29.78	1.00
6	12.9	67.4	29.39	1.10
7	8.3	66.8	29.69	1.15
8	20.1	76.9	29.48	1.03
9	72.2	77.7	29.09	0.77
10	24.0	67.7	29.60	1.07
11	23.2	76.8	29.38	1.07
12	47.4	86.6	29.35	0.94
13	31.5	76.9	29.63	1.10
14	10.6	86.3	29.56	1.10
15	11.2	86.0	29.48	1.10
16	73.3	76.3	29.40	0.91
17	75.4	77.9	29.28	0.87
18	96.6	78.7	29.29	0.78
19	107.4	86.8	29.03	0.82
20	54.9	70.9	29.37	0.95
21				

Persamaan regresi dan hasil taksiran nilai fungsi:

```
Persamaan regresi:  
f(x) = -3.5078 + -0.0026x1 + 0.0008x2 + 0.1542x3  
Taksiran nilai fungsi:  
f(xk) = 0.9384  
  
Apakah ingin menyimpan hasil? (Y/N)  
y  
Masukkan nama file (.txt)  
output5.txt
```

Diperoleh persamaan regresi adalah  $f(x) = -3.5078 - 0.0026x_1 - 0.0008x_2 + 0.1542x_3$

Diperoleh estimasi nilai Nitrous Oxide apabila Humidity bernilai 50%, temperatur 76°F, dan tekanan udara sebesar 29.30 adalah 0.9384

## **BAB 5**

### **Kesimpulan, Saran, dan Refleksi**

#### **I. Kesimpulan**

Program ini dapat digunakan untuk perhitungan seperti :

1. Menghitung solusi Sistem Persamaan Linier dengan metode Eliminasi Gauss, Eliminasi Gauss-Jordan, Balikan Matriks, dan Kaidah Cramer.
2. Menghitung determinan sebuah matriks dengan metode Eliminasi Gauss dan Ekspansi Kofaktor
3. Menghitung matriks balikan, dan
4. Menghitung fungsi dan hasil taksiran pada persoalan Interpolasi Polinom, Bikubik, dan Regresi Linier Berganda

#### **II. Saran**

Pada pengerjaan tugas besar ini, terjadi beberapa miskomunikasi antara anggota kelompok seperti penggunaan tipe data dalam program. Kemudian, ada pula kendala waktu karena pada awal perilsan tugas, pengerjaan ditunda selama beberapa hari sehingga waktu efektif untuk mengerjakan menjadi lebih sedikit. Oleh karena itu, disarankan untuk sebelum mengerjakan tugas, tentukan secara rinci dasar-dasar program yang akan digunakan sehingga tidak terjadi kesalahan yang mempersulit pengerjaan tugas nanti. Juga agar waktu mengerjakan tugas cukup, mulailah sesegera mungkin sebab kita tidak tahu masalah seperti apa yang akan ditemui dan seberapa lama waktu yang dibutuhkan untuk menyelesaikan permasalahan tersebut.

#### **III. Refleksi**

Pada tugas besar ini, kami belajar bahwa waktu yang tersedia sangat berharga dan jangan menunda-nunda pekerjaan sehingga dapat selesai dengan tepat waktu tanpa merasa tertekan karena mendekati deadline. Juga kami belajar bahwa kolaborasi dan komunikasi antar anggota itu sangatlah penting untuk keberjalanan pembuatan projek. Kami juga belajar untuk mengelola repository pada github.com untuk pengerjaan tugas coding seperti ini.

*Repository github*

<https://github.com/HoseaNA/Algeo01-21057.git>

## Referensi

1. <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/algeo22-23.htm>
2. [Inverse matrices for linear equations with infinite solutions - Mathematics Stack Exchange](#)
3. [Lesson Explainer: Cramer's Rule | Nagwa](#)
4. [5.3 - The Multiple Linear Regression Model | STAT 462 \(psu.edu\)](#)
5. [5.4 - A Matrix Formulation of the Multiple Regression Model | STAT 462 \(psu.edu\)](#)
6. [Multiple Linear Regression by Hand \(Step-by-Step\) - Statology](#)
7. <https://www.statlect.com/matrix-algebra/Gauss-Jordan-elimination>
8. <https://engineering.ucsb.edu/~hpscicom/projects/gauss/introge.pdf>