Tugas Besar 1 IF 2123 Aljabar Linier dan Geometri

Sistem Persamaan Linier, Determinan, dan Aplikasinya

Anggota Kelompok Triplek :

- Raynard Tanadi : 13521143
- Marcel Ryan : 13521127
- Kenneth Dave Bahana : 13521145

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

2022

BAB 1
DESKRIPSI MASALAH

A. Gambaran Umum

Mahasiswa IF 2123 diminta untuk membuat pustaka dalam **Bahasa Java** untuk menemukan solusi SPL dengan metode eliminasi Gauss, metode Eliminasi Gauss-Jordan, metode matriks balikan, dan kaidah Cramer (kaidah Cramer khusus untuk SPL dengan *n* peubah dan *n* persamaan), menghitung determinan matriks dengan reduksi baris dan dengan ekspansi kofaktor, dan menghitung balikan matriks.

B. Spesifikasi Detail

Gunakan pustaka di atas untuk membuat program penyelesaian berbagai persoalan dalam bentuk SPL, menyelesaikan persoalan interpolasi dan regresi linier, menghitung matriks balikan, menghitung determinan matriks dengan berbagai metode (reduksi baris dan ekspansi kofaktor). Sepsifikasi program 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 a_{ij} , dan b_i . Masukan dari *file* berbentuk matriks *augmented* tanpa tanda kurung, setiap elemen matriks dipisah oleh spasi. Misalnya,

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

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

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

3. Untuk persoalan interpolasi, masukannya jika dari *keyboard* adalah n, (x_0 , y_0), (x_1 , y_1), ..., (x_n , y_n), dan nilai x yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung. 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 x_{1i} , x_{2i} , ..., x_{ni} , nilai y_i , dan nilai-nilai x_k yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung.

5. Untuk persoalan SPL, luaran (*output*) program adalah solusi SPL. Jika solusinya tunggal, tuliskan nilainya. Jika solusinya tidak ada, tuliskan solusi tidak ada, jika solusinya banyak, maka tuliskan solusinya dalam bentuk parametrik (misalnya x_4 = -2, x_3 = $2s - t$, $x_2 = s$, dan $x_1 = t$.)

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

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

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

8. Untuk persoalan interpolasi bicubic, masukan dari file text (.txt) yang berisi matriks berukuran 4x4 yang berisi nilai f(i,j) dengan i dan j adalah indeks matriks diikuti dengan nilai a dan b untuk mencari nilai f(a,b). misalnya jika 6 nilai dari f(-1,-1), f(-1,0), f(-1,1), f(-1,2),f(0,-1), f(0,0), f(0,1), f(0,2), f(1,-1), f(1,0), f(1,1), f(1,2), f(2,-1), f(2,0), f(2,1), f(2,2) berturut-turut adalah 1, 2, 3, 4, 5, 6, 7, 8, 9 ,10, 11, 12, 13, 14, 15 ,16 serta nilai a dan b yang dicari berturut-turut adalah 0.5 dan 0.5 maka isi file text ditulis sebagai berikut:

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

   luaran yang dihasilkan adalah nilai dari f(0.5,0.5).
   masukannya adalah matriks 4 x 4, diikuti oleh nilai a dan b, maka luarannya adalah nilai $f(a,b)$.

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

10. Bahasa program yang digunakan adalah Java.
11. Program **tidak harus** berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas *Eclipse* misalnya).
12. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

    MENU
    1. Sistem Persamaan Linier
    2. Determinan
    3. Matriks balikan
    4. Interpolasi Polinom
    5. Interpolasi Bicubic
    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

A. Metode Eliminasi Gauss

Eliminasi Gauss adalah metode yang ditemukan oleh Carl Friedrich Gauss, metode ini dapat digunakan untuk memecahkan sistem persamaan linear dengan merepresentasikan menjadi bentuk matriks, lalu matriks tersebut diubah menjadi bentuk Eselon Baris dengan Operasi Baris Elementer, dan sistem diselesaikan dengan substitusi balik. Eselon baris memiliki tiga kriteria, yaitu jika elemen dari suatu baris tidak semuanya nol, maka bilangan tak nol pertama di dalam baris adalah 1, lalu kalau ada baris yang semua elemennya bernilai 0, maka baris tersebut harus dikelompokkan dan diletakkan di bagian bawah matriks, dan jika terdapat dua baris berurutan yang memenuhi kriteria pertama, maka angka 1 dari baris yang lebih rendah berada lebih ke kanan dari angka 1 baris yang ada di atasnya. Solusi yang didapat dari metode ini terdapat tiga jenis, yaitu memiliki tepat satu solusi, solusi banyak, dan tidak memiliki solusi sama sekali. Matriks memiliki tepat satu solusi jika dua kolom terakhir tidak bernilai nol. Lalu, matriks memiliki solusi banyak jika ada baris yang semua elemennya nol atau jumlah baris lebih sedikit dibanding jumlah kolom dikurang satu, matriks jenis ini akan memiliki solusi akhir berbentuk parametrik. Terakhir, matriks tidak memiliki solusi sama sekali jika ada baris yang semua elemennya nol kecuali kolom terakhir.

B. Metode Eliminasi Gauss-Jordan

Eliminasi Gauss-Jordan adalah metode pemecahan sistem persamaan linear dengan mengubahnya menjadi bentuk matriks eselon baris tereduksi dengan menggunakan operasi baris elementer. Matriks eselon baris tereduksi sendiri adalah matriks eselon baris yang lebih disederhanakan untuk mempermudah pencarian solusi dari suatu sistem persamaan. Eselon bairs tereduksi memliki empat kriteria, yaitu ketiga kriteria eselon baris dan setiap kolom yang mengandung 1 utama maka elemen elemen lain selain 1 utama harus bernilai 0. Metode ini akan lebih terasa bermanfaat jika sistem persamaan linear yang dipecahkan memiliki jumlah persamaan dan variabel yang sama.

C. Determinan

Determinan adalah nilai skalar yang dapat dihitung dari suatu matriks persegi. Determinan matriks A dapat ditulis sebagai det(A), det A, atau $|A|$. Determinan dapat digunakan untuk membantu menemukan matriks balikan. Determinan dapat didapatkan dengan beberapa cara, seperti reduksi baris yang menggunakan operasi baris elementer pada matriks sampai diperoleh matriks segitiga yang setelah itu semua elemen diagonal dari matriks tersebut dikalikan dan ekspansi kofaktor yang menjumlahkan perkalian elemen dengan elemen kofaktor matriks yang sesuai.

D. Matriks Balikan

Matriks balikan adalah matriks yang jika dikali dengan matriks lain, maka akan menghasilkan matriks identitas. Matriks balikan dari A dapat ditulis sebagai A^{-1}. Tidak semua matriks memiliki matriks balikan karena tidak semua matriks memiliki konfigurasi matriks untuk dikalikan dan menghasilkan matriks identitas. Matriks yang determinannya nol atau baris dan kolomnya berbeda, maka tidak akan memiliki matriks balikan. Matriks yang tidak memiliki balikan disebut matriks singular. Matriks balikan sendiri dapat didapatkan dengan beberapa cara, seperti dengan menggunakan metode eliminasi Gauss-Jordan dan dengan menggunakan rumus berikut :

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

E. Matriks Kofaktor

Matriks kofaktor adalah matriks yang terdiri dari kofaktor-kofaktor itu sendiri. Kofaktor sendiri merupakan hasil perkalian minor dengan $(-1)^{i+j}$ dimana i adalah baris dan j adalah kolom. Kofaktor dari elemen baris ke-i dan kolom ke-j ditulis sebagai C_{ij}. Selain itu, minor sendiri adalah determinan matriks bagian dari matriks utama yang diperoleh dengan cara menghilangkan elemen pada baris ke-i dan elemen kolom ke-j. Minor dari dari baris ke-i dan kolom ke-j ditulis sebagai M_{ij}. Jika matriks kofaktor ditranspose, maka akan dihasilkan matriks Adjoin yang dapat digunakan untuk mencari matriks balikan nantinya. Matriks kofaktor juga dapat berguna untuk menghitung determinan dengan menggunakan metode ekspansi kofaktor yang menjumlahkan perkalian elemen dengan elemen kofaktor matriks yang sesuai.

F. Matriks Adjoin

Matriks adjoin adalah transpose dari matriks kofaktor dan biasa matriks adjoin A ditulis sebagai Adj (A). Matriks adjoin dapat berguna dalam mencari matriks balikan yang juga memerlukan determinan sesuai dengan rumus berikut :

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

G. Kaidah Cramer

Kaidah Cramer ditemukan oleh Gabriel Cramer dan biasa digunakan untuk menyelesaikan sistem persamaan linear dengan variabel banyak yang berbentuk Ax = b dan hanya berlaku jika sistem persamaan linear tersebut memiliki solusi yang tunggal dan determinan tidak nol. Contoh solusi unik yang akan dimiliki adalah sebagai berikut :

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

Dalam hal ini, A_n adalah matriks yang diperoleh dengan mengganti entri pada kolom ke-n dari A dengan entri dari matriks berikut :

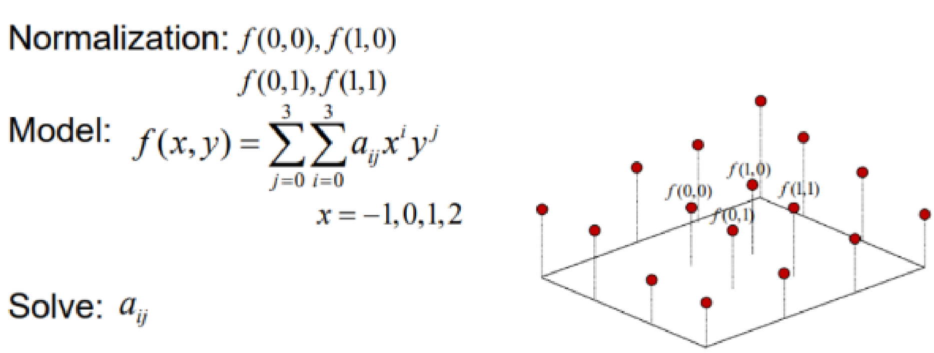
$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

H. Interpolasi Polinom

Interpolasi polinom merupakan teknik interpolasi yang mengasumsikan pola data yang dimiliki mengikuti pola polinom linier maupun berderajat tinggi untuk memperkirakan suatu nilai. Persoalan interpolasi polinom adalah sebagai berikut: Diberikan n+1 buah titik berbeda, $(x_0\ ,\ y_0)$, $(x_1\ ,\ y_1)$, ..., $(x_n\ ,\ y_n)$. Tentukan polinom $p_n(x)$ yang menginterpolasi (melewati) semua titik-titik tersebut sedemikian rupa sehingga $y_i = p_n(x_i)$ untuk i = 0, 1, 2, …, n. Persamaan polinom yang didapatkan akan berbentuk $p_n(x) = a_0 + a_1\mathrm{x} + a_2x^2 + \ldots + a_nx^n$. Lalu, untuk mendapatkan polinom interpolasi $p_n(x)$, diperlukan penyulihan balik titik $(x_1\ , y_1)$ ke dalam persamaan $p_n(x) = a_0 + a_1\mathrm{x} + a_2x^2 + \ldots + a_nx^n$ untuk i=0, 1, 2, …, n dan melakukan eliminasi Gauss. Setelah polinom interpolasi $p_n(x)$ ditemukan, $p_n(x)$ dapat digunakan untuk menghitung perkiraan nilai y di sembarang titik di dalam selang $[x_0\ , x_n]$.

I. Interpolasi Bicubic

Bicubic interpolation merupakan teknik interpolasi pada data 2D umumnya digunakan dalam pembesaran citra yang merupakan pengembangan dari interpolasi linear. Suatu *image* dengan menggunakan teknik interpolasi data 2D ini, dapat diperbesar dengan menggunakan persamaan hasil dari interpolasi. Contoh representasi dari suatu image tersebut berupa matriks piksel dari foto tersebut. Diberikan sebuah matrix awal, misal M, kita akan mencari persamaan interpolasi f(x,y) dengan pemodelan sebagai berikut:



Melakukan substitusi nilai-nilai diketahui pada matriks 4 x 4 tersebut ke persamaan f(x,y) untuk diregresikan dan mendapatkan banyak pemetakan nilai f(x,y) dari x -1 sampai 2 yang berbentuk bilangan riil. Salah satu yang dapat diimplementasikan untuk menghasilkan matriks pembesaran 2 kali lipat (16 x 16) adalah untuk mencari nilai tengah dari pemetakan matriks yang sudah ada. Misalnya f(0.5, 0.5). Untuk mendapatkan nilai tersebut, matriks 4x4 tadi dapat di substitusikan ke persamaan f(x,y) sebagai berikut:

$$y = Xa$$

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

Persamaan tersebut berbentuk serangkaian persamaan f(x,y) berupa:

$f(-1,-1) = a_{00} + a_{10}.(-1)^1 + a_{11}(-1)^1(-1)^1 + a_{20}(-1)^2 + ... + a_{33}(-1)^3(-1)^3$

$f(0,-1) = a_{00} + a_{10}.(0)^1 + a_{11}(0)^1(-1)^1 + a_{20}(0)^2 + ... + a_{33}(0)^3(-1)^3$

$f(2,2) = a_{00} \ldots.. + a_{33}(x)^3(y)^3$

Ket: n representasi jumlah baris dan y jumlah kolom matriks.

Persamaan yang didapatkan akan menghasilkan semua nilai a_{ij} yang akhirnya didapatkan persamaan akhir f(x,y) dan bisa digunakan untuk mencari nilai antara data matriks yang sudah ada seperti f(0.5, 0.5), f(1.5, 1.5) dan lainnya diantara -1 sampai 2 pada contoh. Hasil dari f(x,y) tersebut dapat digunakan untuk “memperbesar” matriks sehingga bisa digunakan untuk memperbesar image pada pengaplikasiannya.

J. Regresi Linier Berganda

Regresi linear berganda dapat digunakan untuk meprediksi nilai seperti interpolasi polinom. Rumus umum dari regresi linear yang bisa digunakan untuk regresi linear berganda adalah sebagai berikut :

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

Untuk mendapatkan nilai setiap β, digunakan *Normal Estimation Equation for Multiple Linear Regression* sebagai berikut :

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

Terakhir, sistem persamaan linier diselesaikan dengan menggunakan metode eliminasi Gauss.

BAB 3
IMPLEMENTASI PROGRAM

1. Class Matrix

- Attribute

Nama	Tipe	Deskripsi
baris	Protected int	Jumlah baris
kolom	Protected int	Jumlah kolom
matrix	Public double [][]	Elemen-elemen di dalam matriks
count	Protected int	Menghitung jumlah swap antar kolom

- Fungsi-fungsi

Nama	Tipe	Parameter	Deskripsi
Matrix	Public Constructor	Int row Int col Double[][] matrix	Membuat tipe variabel baru bertipe Matrix yang berisi int dan double[][]
readMatrix	Public void		Mengset nilai-nilai ke dalam variabel Matrix
printMatrix	Public void		Mengeluarkan output berupa elemen-elemen di dalam matrix
minusBaris	Public void	Double isi Int m Int n	Operasi pengurangan baris yaitu baris1 - isi * baris2 (untuk membantu pencarian matriks gauss)
bagiMatriks	Public void	Int m Double k	Membagi seluruh elemen matriks pada baris m dengan double k

baris0Semua	Public boolean	Int m	Mengembalikan true apabila 1 baris berisi 0 semua
kolom0Semua	Public boolean	Int m	Mengembalikan true apabila 1 kolom berisi 0 semua
isiBukan0	Public int	Int m	Mengembalikan integer pertama yang bukan 0 di suatu baris
susunMatrix	Public void	Int m1 Int m2	Menyusun barisan matriks agar baris dengan 0 terbanyak berada di paling bawah dan 0 paling sedikit berada di paling atas
tukarBaris	Public void	Int m1 Int m2	Menukarkan posisi 2 baris di dalam suatu matrix
gauss	Public void		Melakukan operasi gauss kepada matriks
pembagiGauss	Public void		Membagi semua nilai gauss agar nilai pada index kolom dan baris 1 utama menjadi 1
gaussJordan	Public void		Melakukan operasi Gauss Jordan kepada matriks
determinan	Public double		Mencari determinan dari sebuah matriks
copyMatrix	Public Matrix	Matrix mbaru Matrix matriks	Mengembalikan matrix mbaru yang isinya berupa salinan dari matrix matriks
isSquare	Public boolean	Int rowEff Int colEff	Mengembalikan true apabila baris dan

			kolom matriks sama
inverseOBENoLine	Public void		Melakukan operasi invers kepada matriks dan mengembalikan matriks yang telah di invers tanpa print string
inverseOBE	Public void		Melakukan operasi invers kepada matriks dan mengembalikan matriks yang telah di invers
kaidahCramer	Public void		Melakukan operasi kaidah Cramer kepada matriks untuk mencari solusi dari persamaan matriks tersebut
persamaanKoefisien	Public Matrix	Matrix mat Int bar Int kol	Menghasilkan matrix 16x16 koefisien x sebagai persamaan y = Xa dari input matriks interpolasi bikubik
inverseKofaktor	Public Matrix		Menghasilkan inverse matrix suatu input dengan metode ekspansi kofaktor.
adjoin	Public Matrix		Menghasilkan adjoin dari suatu input matrix menggunakan minor kofaktor dan transpose
transpose	Public Matrix		Mengembalikan matrix dalam keadaan sudah di transpose (baris dan kolom ditukar)
determinanKofaktor	Public double		Mengembalikan

			determinan matrix dengan menggunakan metode Kofaktor
minorkofaktor	Public Matrix		Menghasilkan minor kofaktor suatu inputan matrix.
ekspansi	Public Matrix	Int tbar Int tkol	Menghasilkan matriks yang lebih kecil sebagai ekspansi dari kofaktor perhitungan determinan metode ekspansi kofaktor.
ambilKol1Utama	Public int[]	Matrix mat	Mengambil index index yang merupakan index 1 utama
isKol1Utama	Public boolean	Int[] kol Int panjang Int x	Mengembalikan true apabila index merupakan index kolom 1 utama
printSolusi	Public void		Memberi output berupa hasil dari persamaan matriks yang diinput dalam bentuk persamaan
perkalianMatriks	Static Matrix	Matrix m1 Matrix m2	Mengembalikan matrix baru yang berupa hasil perkalian dari matrix m1 dan m2
SPLinversOBELine	Public String		Mengembalikan solusi dalam bentuk string dari matrix dengan menggunakan metode invers
SPLinversOBE	Public Matrix		Mengembalikan solusi dari matrix dengan menggunakan

			metode invers
FileBaris	Public static int	String NamaFile	Mengembalikan jumlah baris dari matrix pada file txt yang dibaca
FileKolom	Public static int	String NamaFile	Mengembalikan jumlah kolom dari matrix pada file txt yang dibaca
ReadMatrixFile	Public void	String NamaFile	Membaca dan mengubah elemen matrix sesuai dengan matrix pada file txt yang dibaca
outputFileMatriks	Public void		Menulis file txt yang berisi solusi dalam bentuk matrix
outputFileLine	Public void	String line	Menulis file txt yang berisi solusi dalam bentuk String
outputFileNumber	Public void	Double det	Menulis file txt yang berisi solusi dalam bentuk Double
konfirmasiOutputFile	Public void	Int tipe Double det String line	Melakukan konfirmasi apakah solusi ingin disimpan ke sebuah file. Mengakses outputFile jika solusi ingin disimpan ke sebuah file
solusiGauss	Public Matrix		Mengembalikan solusi dari persamaan matriks dalam bentuk Matrix

2. Class RegresiLinear

- Atribut

Nama	Tipe	Deskripsi
-	-	-

- Fungsi-fungsi

Nama	Tipe	Parameter	Deskripsi
RegresiLinear	Public constructor	Int m Int n	Mengambil baris dan kolom dari Matrix
penjumlahanKolom	Public static double	Matrix mat Int n	Mengembalikan nilai penjumlahan seluruh nilai di 1 kolom
perkalianKolom	Public static Matrix	Matrix mat Int n1 Int n2	Mengembalikan Matrix hasil dari perkalian antar 2 kolom matrix
kaliKolomBedaMatrix	Public static Matrix	Matrix mat1 Matrix mat2 Int n1 Int n2	Mengembalikan matrix yang merupakan hasil perkalian kolom antar matrix1 dan matrix2
printSPL	Public void		Memberikan output berupa SPL yang didapat untuk mencari nilai Beta pada regresi
printPersamaan	Public String		Mengembalikan nilai string yang isinya merupakan hasil persamaan dari regresi nilai berganda
regresiBerganda	Public void		Melakukan operasi regresi linear berganda pada data yang diinput
FileBarisRegresi	Public static int	String NamaFile	Mengembalikan jumlah baris dikurang 3 dari matrix pada file txt yang dibaca

FileKolomRegresi	Public static int	String NamaFile	Mengembalikan jumlah kolom dari matrix pada file txt yang dibaca yang dilongkap dua baris
FileKolomPengubah	Public static int	String NamaFile	Mengembalikan jumlah kolom dari matrix pada file txt yang dibaca yang dilongkap (FileBarisRegresi+2) baris
ReadRegresiFile	Public void	String NamaFile	Membaca dan mengubah elemen matrix sesuai dengan matrix pada file txt yang dibaca yang dilongkap dua baris dan hanya sebanyak (FileBarisRegresi) baris
ReadPengubahFile	Public void	String NamaFile	Membaca dan mengubah elemen matrix sesuai dengan matrix pada file txt yang dibaca yang dilongkap (FileBarisRegresi +2) baris

3. Class Menu

- Atribut

Nama	Tipe	Deskripsi
-	-	-

- Fungsi–fungsi

Nama	Tipe	Parameter	Deskripsi
menuUtama	Public static int		Mengembalikan integer yang berupa

			pilihan menu dari user
menuSPL	Public static int		Mengembalikan integer yang berupa pilihan cara penyelesaian SPL dari user
menuDeterminan	Public static int		Mengembalikan integer yang berupa pilihan cara untuk mendapatkan determinan matrix dari user
menuInput	Public static int		Mengembalikan integer yang berupa pilihan input matrix dari user
menuInvers	Public static int		Mengembalikan integer yang berupa pilihan cara untuk mendapatkan invers dari matrix dari user
runMenu	Public static void		Menjalankan program Matrix-matrix yang telah dibuat

4. Class Interpolasi

- Atribut

Nama	Tipe	Deskripsi
-	-	-

- Fungsi-fungsi

Nama	Tipe	Parameter	Deskripsi
Interpolasi	Public constructor	Int m Int n	Mengambil baris dan kolom dari

			Matrix
interpolasi	Public double	Interpolasi arraytitik int pilihan	Mengembalikan solusi interpolasi dengan operasi interpolasi polinom
interpolasiline	Public string	Interpolasi arraytitik int pilihan	Mengembalikan solusi interpolasi dalam bentuk string dengan operasi interpolasi polinom
FileBarisInterpolasi	Public static int	String NamaFile	Mengembalikan jumlah baris dikurang 2 dari matrix pada file txt yang dibaca
FileKolomInterpolasi	Public static int	String NamaFile	Mengembalikan jumlah kolom dari matrix pada file txt yang dibaca yang dilongkap satu baris
ReadInterpolasiTitikFile	Public void	String NamaFile	Membaca dan mengubah elemen matrix sesuai dengan matrix pada file txt yang dibaca yang dilongkap (FileBarisInterpolasi + 1) baris
ReadInterpolasiFile	Public void	String NamaFile	Membaca dan mengubah elemen matrix sesuai dengan matrix pada file txt yang dibaca yang dilongkap satu baris dan sebanyak

			(FileBarisInterpolasi) baris

5. Class InterpolasiBikubik

- Atribut

Nama	Tipe	Deskripsi
-	-	-

- Fungsi–fungsi

Nama	Tipe	Parameter	Deskripsi
inputfxy	Public Matrix	Matrix matrixsolusi	Memetakan ulang input nilai - nilai f(x,y) interpolasi bikubik menjadi sebuah matrix dengan jumlah kolom 1. Berfungsi untuk dikalikan dengan matrix inverse koefisien.
persamaanKoefisien	Public Matrix	Matrix exp Double xi Double yj	Mendapatkan persamaan koefisien dari $x^i y^j$ (a_{ij})
inverseX	Public InterpolasiBikubik		Mengembalikan inverse dari matrix, dengan spesifikasi matrix persamaanKoefisien.
koefisien	Public Matrix	InterpolasiBikubik koef	Merupakan perkalian antara solusi f(x,y), yaitu matrix dalam inputfxy dengan inverseX, yaitu inverse persamaan koefisien menghasilkan nilai - nilai koefisien a_{ij}.
fxy	Public double	InterpolasiBikubik	Menyimpan dan

		titik InterpolasiBikubik koef Double hasil	membentuk persamaan f(x,y) dari koefisien a_{ij} yang telah dihasilkan dan meminta input nilai x,y untuk mendapatkan suatu hasil f(x,y).
ReadBicubicFile	Public void	String NamaFile	Membaca matrix inputan setiap f(x,y) dari suatu file txt.
ReadBicubicArrayFile	Public void	String NamaFile	Membaca nilai input nilai x dan y yang akan digunakan ke dalam persamaan f(x,y) yang sudah jadi.

BAB 4
EKSPERIMEN

1. Solusi SPL Ax = b

a	Kaidah Cramer
$A = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 2 & 5 & -7 & -5 \\ 2 & -1 & 1 & 3 \\ 5 & 2 & -4 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ -2 \\ 4 \\ 6 \end{bmatrix}$	Untuk SPL dengan metode kaidah cramer, Mohon input baris dan kolom dengan kolom = baris+1 Masukkan baris: 4 Masukkan kolom: 5 1 1 -1 -1 1 2 5 -7 -5 -2 2 -1 1 3 4 5 2 -4 2 6 Metode ini tidak bisa digunakan untuk persamaan matrix tersebut.

b	Metode Gauss-Jordan
$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 0 & -3 & 0 \\ 2 & -1 & 0 & 1 & -1 \\ -1 & 2 & 0 & -2 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 6 \\ 5 \\ -1 \end{bmatrix}$	Solusi yang didapat dari persamaan adalah: x1 = 3.0 + w x2 = + 2.0w x4 = -1.0 + w x5 = w

c	Metode Gauss
$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$	Solusi yang didapat dari persamaan adalah: x2 = 1.0 - x x4 = -2.0 - x x5 = 1.0 + x x6 = x

d	Metode Invers OBE
$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n+1} \end{bmatrix},\ b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$ *H adalah matriks Hilbert. Cobakan untuk n = 6 dan n = 10.* Jika n = 10	Hasil SPL adalah : X[1] = 8.444 X[2] = -23.812 X[3] = -22.079 X[4] = 47.749 X[5] = 34.292 X[6] = -43.706 X[7] = 0.880 X[8] = 30.544 X[9] = 0.012 X[10] = -36.273

e	Kaidah Cramer
$-x + 2y - 3z = 1$ $2x + z = 0$ $3x - 4y + 4z = 2$	Berikut solusi yang didapat dari kaidah cramer : x1 = 0.8, x2 = -1.5, x3 = -1.6

f	Metode Gauss-Jordan
$x_1 - x_2 + 2x_3 - x_4 = -1$ $2x_1 + x_2 - 2x_3 - 2x_4 = -2$ $-x_1 + 2x_2 - 4x_3 + x_4 = 1$ $3x_1 - 3x_4 = -3$	Solusi yang didapat dari persamaan adalah: x1 = -1.0 + v x2 = + 2.0u x3 = u x4 = v

g	Metode Gauss

$2x_1 + 3x_2 - x_3 = 5$ $4x_1 + 4x_2 - 3x_3 = 3$ $-2x_1 + 3x_2 - x_3 = 1$	Berikut solusi yang didapatkan dari persamaan matrix diatas : Nilai x1 = 1.0 Nilai x2 = 2.0 Nilai x3 = 3.0

h	**Metode Gauss**
$x_1 + 3x_2 - 2x_3 + 2x_5 = 0$ $2x_1 + 6x_2 - 5x_3 - 2x_4 + 4x_5 - 3x_6 = -1$ $5x_3 + 10x_4 + 15x_6 = 5$ $2x_1 + 6x_2 + 8x_4 + 4x_5 + 18x_6 = 6$	Solusi yang didapat dari persamaan adalah: x1 = - 2.0w - 4.0v - 3.0t x2 = t x3 = - 2.0v x4 = v x5 = w x6 = 0.3333333333333333

i (Metode Gauss)
$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n+1} \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

Jika n = 6	Berikut solusi yang didapatkan dari persamaan matrix diatas : Nilai x1 = 8.080835703347583 Nilai x2 = -23.55884926712259 Nilai x3 = -31.735397068466554 Nilai x4 = 108.01793918179212 Nilai x5 = -43.69940931962219 Nilai x6 = -17.952854955152787

2. SPL berbentuk matriks *augmented*

a	**Metode Gauss-Jordan**

a. $$\begin{bmatrix} 1 & -1 & 2 & -1 & -1 \\ 2 & 1 & -2 & -2 & -2 \\ -1 & 2 & -4 & 1 & 1 \\ 3 & 0 & 0 & -3 & -3 \end{bmatrix}.$$	Solusi yang didapat dari persamaan adalah: x1 = -1.0 + v x2 = + 2.0u x3 = u x4 = v

b	Metode Gauss
b. $$\begin{bmatrix} 2 & 0 & 8 & 0 & 8 \\ 0 & 1 & 0 & 4 & 6 \\ -4 & 0 & 6 & 0 & 6 \\ 0 & -2 & 0 & 3 & -1 \\ 2 & 0 & -4 & 0 & -4 \\ 0 & 1 & 0 & -2 & 0 \end{bmatrix}.$$	Tidak ada solusi untuk matrix ini

3. SPL berbentuk

a	Kaidah Cramer
$$\begin{aligned} 8x_1 + x_2 + 3x_3 + 2x_4 &= 0 \\ 2x_1 + 9x_2 - x_3 - 2x_4 &= 1 \\ x_1 + 3x_2 + 2x_3 - x_4 &= 2 \\ x_1 + 6x_3 + 4x_4 &= 3 \end{aligned}$$	Berikut solusi yang didapat dari kaidah cramer : x1 = -0.22432432432432428 x2 = 0.1824324324324324 x3 = 0.7094594594594594 x4 = -0.25810810810810797

<table>
<tr><th>b</th><th>Metode Gauss Jordan</th></tr>
<tr><td>b.
$$\begin{aligned} x_7 + x_8 + x_9 &= 13.00 \\ x_4 + x_5 + x_6 &= 15.00 \\ x_1 + x_2 + x_3 &= 8.00 \\ 0.04289(x_3 + x_5 + x_7) + 0.75(x_6 + x_8) + 0.61396x_9 &= 14.79 \\ 0.91421(x_3 + x_5 + x_7) + 0.25(x_2 + x_4 + x_6 + x_8) &= 14.31 \\ 0.04289(x_3 + x_5 + x_7) + 0.75(x_2 + x_4) + 0.61396x_1 &= 3.81 \\ x_3 + x_6 + x_9 &= 18.00 \\ x_2 + x_5 + x_8 &= 12.00 \\ x_1 + x_4 + x_7 &= 6.00 \\ 0.04289(x_1 + x_5 + x_9) + 0.75(x_2 + x_6) + 0.61396x_3 &= 10.51 \\ 0.91421(x_1 + x_5 + x_9) + 0.25(x_2 + x_4 + x_6 + x_8) &= 16.13 \\ 0.04289(x_1 + x_5 + x_9) + 0.75(x_4 + x_8) + 0.61396x_7 &= 7.04 \end{aligned}$$</td><td>Tidak ada solusi untuk matrix ini</td></tr>
</table>

4. Studi Kasus Interpolasi

<table>
<tr><th>a1</th><th>Interpolasi Polinom</th></tr>
<tr><td><table><tr><td>x</td><td>0.4</td><td>0.7</td><td>0.11</td><td>0.14</td><td>0.17</td><td>0.2</td><td>0.23</td></tr><tr><td>f(x)</td><td>0.043</td><td>0.005</td><td>0. 058</td><td>0.072</td><td>0.1</td><td>0.13</td><td>0.147</td></tr></table>
Jika nilai x = 0.2</td><td>P2(x) = -0.18455901923009l6 + 10.27638399147065x^1 + -163.915662634771
76x^2 + 1220.8548907799961x^3 + -4346.313951307695x^4 + 7102.399163251
57x^5 + -4212.4345322103345x^6
Hasil interpolasi adalah : 0.130</td></tr>
</table>

<table>
<tr><th>a2</th><th>Interpolasi Polinom</th></tr>
<tr><td><table><tr><td>x</td><td>0.4</td><td>0.7</td><td>0.11</td><td>0.14</td><td>0.17</td><td>0.2</td><td>0.23</td></tr><tr><td>f(x)</td><td>0.043</td><td>0.005</td><td>0. 058</td><td>0.072</td><td>0.1</td><td>0.13</td><td>0.147</td></tr></table>
Jika nilai x = 0.55</td><td>P2(x) = -0.1845590192300916 + 10.27638399147065x^1 + -163.915662634771
76x^2 + 1220.8548907799961x^3 + -4346.313951307695x^4 + 7102.399163251
57x^5 + -4212.4345322103345x^6
Hasil interpolasi adalah : 2.138</td></tr>
</table>

<table>
<tr><th>a3</th><th>Interpolasi Polinom</th></tr>
<tr><td><table><tr><td>x</td><td>0.4</td><td>0.7</td><td>0.11</td><td>0.14</td><td>0.17</td><td>0.2</td><td>0.23</td></tr><tr><td>f(x)</td><td>0.043</td><td>0.005</td><td>0. 058</td><td>0.072</td><td>0.1</td><td>0.13</td><td>0.147</td></tr></table>
Jika nilai x = 0.85</td><td>P2(x) = -0.1845590192300916 + 10.27638399147065x^1 + -163.915662634771
76x^2 + 1220.8548907799961x^3 + -4346.313951307695x^4 + 7102.399163251
57x^5 + -4212.4345322103345x^6
Hasil interpolasi adalah : -66.270</td></tr>
</table>

a4	Interpolasi Polinom

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

Jika nilai x = 1.28

```
P2(x) = -0.1845590192300916 + 10.27638399147065x^1 + -163.915662634771
76x^2 + 1220.8548907799961x^3 + -4346.313951307695x^4 + 7102.399163251
57x^5 + -4212.4345322103345x^6
Hasil interpolasi adalah : -3485.145
```

b1	Interpolasi Polinom

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 16/07/2022

```
P2(x) = 7.189658216471531E12 + -9.350004058993242E12x^1 + 5.3357555608
50125E12x^2 + -1.7572765632607043E12x^3 + 3.686407609710408E11x^4 + -5
.114342929271088E10x^5 + 4.696794214320051E9x^6 + -2.7552878113789976E
8x^7 + 9374584.419448344x^8 + -141018.35272207734x^9
Hasil interpolasi adalah : 53537.729
```

b2	Interpolasi Polinom

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 10/08/2022

```
P2(x) = 7.189658216471531E12 + -9.350004058993242E12x^1 + 5.3357555608
50125E12x^2 + -1.7572765632607043E12x^3 + 3.686407609710408E11x^4 + -5
.114342929271088E10x^5 + 4.696794214320051E9x^6 + -2.7552878113789976E
8x^7 + 9374584.419448344x^8 + -141018.35272207734x^9
Hasil interpolasi adalah : 36295.012
```

b3	Interpolasi Polinom

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 05/09/2022

```
P2(x) = 7.189658216471531E12 + -9.350004058993242E12x^1 + 5.3357555608
50125E12x^2 + -1.7572765632607043E12x^3 + 3.686407609710408E11x^4 + -5
.114342929271088E10x^5 + 4.696794214320051E9x^6 + -2.7552878113789976E
8x^7 + 9374584.419448344x^8 + -141018.35272207734x^9
Hasil interpolasi adalah : -667726.141
```

b4

Interpolasi Polinom

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 03/10/2022

```
P2(x) = 7.189658216471531E12 + -9.350004058993242E12x^1 + 5.3357555608
50125E12x^2 + -1.7572765632607043E12x^3 + 3.686407609710408E11x^4 + -5
.114342929271088E10x^5 + 4.696794214320051E9x^6 + -2.7552878113789976E
8x^7 + 9374584.419448344x^8 + -141018.35272207734x^9
Hasil interpolasi adalah : -332806748.156
```

c

Interpolasi Polinom

c. Sederhanakan fungsi

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

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

```
P2(x) = 0.29700000000000076 + 0.34624999999999959x^1 + -0.1005208333333
2651x^2 + -0.007812500000004406x^3 + 0.003255208333342876x^4
Hasil interpolasi adalah : 0.297
```

<table>
<tr>
<td>5
0.4 0.419
0.8 0.507
1.2 0.561
1.6 0.583
2 0.577
0
Pengujian pada x=0</td>
<td></td>
</tr>
</table>

5. Studi Kasus Interpolasi Bicubic

<table>
<tr>
<th>a</th>
<th>Interpolasi Bikubik</th>
</tr>
<tr>
<td>4. Studi Kasus Interpolasi Bicubic

Diberikan matriks input:

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

Tentukan nilai:
f(0,0) = ?
f(0.5, 0.5) = ?
f(0.25, 0.75) = ?
f(0.1, 0.9) = ?

Menentukan nilai f(0, 0)</td>
<td>f(0.000000, 0.000000) = 161.000000</td>
</tr>
</table>

b	Interpolasi Bikubik
4. **Studi Kasus Interpolasi Bicubic** Diberikan matriks input: $\begin{bmatrix} 153 & 59 & 210 & 96 \\ 125 & 161 & 72 & 81 \\ 98 & 101 & 42 & 12 \\ 21 & 51 & 0 & 16 \end{bmatrix}$ Tentukan nilai: f(0,0) = ? f(0.5, 0.5) = ? f(0.25, 0.75) = ? f(0.1, 0.9) = ? Menentukan nilai f(0.5, 0.5)	`f(0.500000, 0.500000) = 97.726560`

c	Interpolasi Bikubik
4. **Studi Kasus Interpolasi Bicubic** Diberikan matriks input: $\begin{bmatrix} 153 & 59 & 210 & 96 \\ 125 & 161 & 72 & 81 \\ 98 & 101 & 42 & 12 \\ 21 & 51 & 0 & 16 \end{bmatrix}$ Tentukan nilai: f(0,0) = ? f(0.5, 0.5) = ? f(0.25, 0.75) = ? f(0.1, 0.9) = ? Menentukan nilai f(0.25 ,0.75)	`f(0.250000, 0.750000) = 82.502070`

d	Interpolasi Bikubik
4. **Studi Kasus Interpolasi Bicubic** Diberikan matriks input: $\begin{bmatrix} 153 & 59 & 210 & 96 \\ 125 & 161 & 72 & 81 \\ 98 & 101 & 42 & 12 \\ 21 & 51 & 0 & 16 \end{bmatrix}$ Tentukan nilai: f(0,0) = ? f(0.5, 0.5) = ? f(0.25, 0.75) = ? f(0.1, 0.9) = ? Menentukan nilai f(0.1, 0.9)	`f(0.100000, 0.900000) = 74.696112`

6. Studi Kasus Regresi Linear Berganda

a

Diberikan sekumpulan data sesuai pada tabel berikut ini.

Table 12.1: Data for Example 12.1

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

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

Gunakan *Normal Estimation Equation for Multiple Linear Regression* untuk mendapatkan regresi linear berganda dari data pada tabel di atas, kemudian

13

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

Dari data-data tersebut, apabila diterapkan *Normal Estimation Equation for Multiple Linear Regression*, maka diperoleh sistem persamaan linear sebagai berikut.

$$20b_0 + 863.1b_1 + 1530.4b_2 + 587.84b_3 = 19.42$$

$$863.1b_0 + 54876.89b_1 + 67000.09b_2 + 25283.395b_3 = 779.477$$

$$1530.4b_0 + 67000.09b_1 + 117912.32b_2 + 44976.867b_3 = 1483.437$$

$$587.84b_0 + 25283.395b_1 + 44976.867b_2 + 17278.5086b_3 = 571.1219$$

```
Berikut adalah persamaan SPL yang diperoleh dari data-data mengggunakan Multiple Linear Regression :
20.000b0 + 863.100b1 + 1530.400b2 + 587.840b3 = 19.420
863.100b0 + 54876.890b1 + 67000.090b2 + 25283.395b3 = 779.477
1530.400b0 + 67000.090b1 + 117912.320b2 + 44976.867b3 = 1483.437
587.840b0 + 25283.395b1 + 44976.867b2 + 17278.509b3 = 571.122

Berikut adalah persamaan yang didapat untuk menentukan nilai Y:
f(x) = -3.508 - 0.003x1 + 0.001x2 + 0.154x3

Dari Regresi Linear Berganda, nilai Y yang didapat adalah 0.938
```

b

Observation Number	Factor 1 (x_{i1})	Factor 2 (x_{i2})	Yield (y_i)
1	41.9	29.1	251.3
2	43.4	29.3	251.3
3	43.9	29.5	248.3
4	44.5	29.7	267.5
5	47.3	29.9	273.0
6	47.5	30.3	276.5
7	47.9	30.5	270.3
8	50.2	30.7	274.9
9	52.8	30.8	285.0
10	53.2	30.9	290.0
11	56.7	31.5	297.0
12	57.0	31.7	302.5
13	63.5	31.9	304.5
14	65.3	32.0	309.3
15	71.1	32.1	321.7
16	77.0	32.5	330.7
17	77.8	32.9	349.0

Untuk x1 = 47 dam x2 = 31

```
Berikut adalah persamaan SPL yang diperoleh dari data-data mengggunakan Multiple Linear Regression :
17.000b0 + 941.000b1 + 525.300b2 = 4902.800
941.000b0 + 54269.980b1 + 29285.970b2 = 276614.420
525.300b0 + 29285.970b1 + 16253.750b2 = 152021.070

Berikut adalah persamaan yang didapat untuk menentukan nilai Y:
f(x) = -153.512 + 1.239x1 + 12.082x2

Dari Regresi Linear Berganda, nilai Y yang didapat adalah 279.261
```

BAB 5
KESIMPULAN, SARAN, DAN REFLEKSI

A. Kesimpulan

Program yang kami buat dibuat dalam bahasa Java dan merupakan kalkulator matrix yang dapat menyelesaikan enam persoalan utama, yaitu sistem persamaan linear, determinan, matriks balikan, interpolasi polinom, interpolasi bicubic, dan regresi linier berganda. Pada sistem persamaan linear berganda terdapat empat metode, yaitu metode eliminasi Gauss, metode eliminasi Gauss-Jordan, metode matriks balikan, dan kaidah Cramer. Lalu, pada determinan terdapat dua metode, yaitu metode ekspansi kofaktor dan metode OBE. Untuk matriks balikan, terdapat dua metode juga, yaitu metode OBE dan metode adjoin.

B. Saran

Setelah melakukan dan menyelesaikan tugas besar pertama ini, terdapat beberapa saran yang mungkin dapat dipertimbangkan dan dapat digunakan untuk tugas besar - tugas besar selanjutnya. Pertama, waktu pengumpulan atau *deadline* dapat diperpanjang beberapa hari. Selain itu, menurut kami, diperlukan adanya asistensi setidaknya sekali atau dua kali untuk mahasiswa mengklarifikasi detil - detil dan mengetahui arah pengerjaan tugas besar sudah benar atau masih kurang tepat.

C. Refleksi

Dengan melakukan dan menyelesaikan tugas besar pertama ini, kami mendapatkan banyak sekali manfaat dan pengalaman baru. Kami dapat mempelajari bahasa Java dan dapat langsung mengimplementasikannya menjadi suatu program. Lalu, kami juga menjadi lebih mempelajari dan mengenal lebih dalam mengenai matriks dan operasi - operasi lainnya, serta dapat mengimplementasikannya menjadi suatu program. Selain menambah *skill* dalam hal pemrograman, kami juga dapat meningkatkan *soft skill* melalui tugas besar pertama ini, seperti kerja sama, *time management,* proses berpikir dalam menghadapi suatu masalah, dan masih banyak lagi. Selain itu, kami juga dapat lebih mengenal diri kami dan saling merefleksikan diri kami untuk kami kembangkan lagi apa yang mungkin masih kurang dari diri kami. Namun, dari proses pengerjaan tugas besar ini, hambatan yang kami alami salah satunya adalah penggabungan kode yang kami kerjakan, dikarenakan library matrix secara keseluruhan yang kami buat terdapat dalam file yang sama dikerjakan oleh semuanya dan tidak menggunakan github. Hal ini menyebabkan unduh ulang file dan merging isi library secara manual yang kurang efisien dibandingkan dengan penggunaan github dengan push and pull dari branch dan tentunya dapat menjadi pembelajaran bagi kami untuk tidak mengulanginya di tugas - tugas berikutnya.

DAFTAR REFERENSI :

Sumber Pustaka:

1. https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2022-2023/algeo22-23.htm
2. https://www.profematika.com/
3. https://www.mssc.mu.edu/~daniel/pubs/RoweTalkMSCS_BiCubic.pdf

Lampiran:

1. Github: https://github.com/Pemuladigital/Algeo-01-21127
2. Jadwal Demo: Selasa, 3 Oktober 2022 Pukul 10.00 -11.00 WIB
3. Asisten Demo: EnterWind