Laporan Tugas Besar

Aljabar dan Geometri IF2123
Kelompok 42

Aditya Bimawan (13519064),

Rhapsodya Piedro Asmorobangun (13519084),

Leo Cardhio Sih Pratama (13519220)

Kelas Mahasiswa (K-4), Jurusan Teknik Informatika

Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung,

Jl. Ganesha no. 10 Bandung, Indonesia, 40132

BAB I - DESKRIPSI MASALAH

Sistem persamaan linier (SPL) $Ax = b$ dengan n peubah (*variable*) dan m persamaan adalah berbentuk

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\
a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\
\vdots \qquad \vdots \qquad\qquad \vdots \quad &\ \ \vdots \\
a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &= b_m
\end{aligned}
$$

yang dalam hal ini x_i adalah peubah, a_{ij} dan b_i adalah koefisien $\in$ R. 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, atau hanya satu (unik/tunggal).

Sebuah matriks M berukuran $n \times n$

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

determinannya adalah

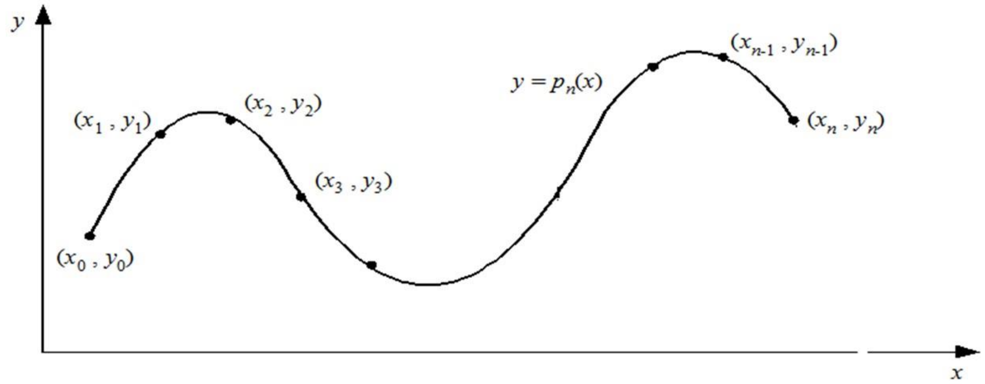
$$
\det(M) = \begin{vmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{vmatrix}
$$

Determinan matriks M berukuran $n \times n$ dapat dihitung dengan beberapa cara: reduksi baris dan ekspansi kofaktor.

SPL memiliki banyak aplikasi dalam bidang sains dan rekayasa, dua diantaranya diterapkan pada tugas besar ini, yaitu interpolasi polinom dan regresi linier.

II. Interpolasi Polinom

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 $i = 0, 1, 2, \dots, n$.



Setelah polinom interpolasi $p_n(x)$ ditemukan, $p_n(x)$ dapat digunakan untuk menghitung perkiraan nilai y di sembarang titik di dalam selang $[x_0, x_n]$.

Polinom interpolasi derajat n yang menginterpolasi titik-titik (x_0, y_0), (x_1, y_1), ..., (x_n, y_n). adalah berbentuk $p_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$. Jika hanya ada dua titik, (x_0, y_0) dan (x_1, y_1), maka polinom yang menginterpolasi kedua titik tersebut adalah $p_1(x) = a_0 + a_1x$ yaitu berupa persamaan garis lurus. Jika tersedia tiga titik, (x_0, y_0), (x_1, y_1), dan (x_2, y_2), maka polinom yang menginterpolasi ketiga titik tersebut adalah $p_2(x) = a_0 + a_1x + a_2x^2$ atau persaman kuadrat dan kurvanya berupa parabola. Jika tersedia empat titik, (x_0, y_0), (x_1, y_1), (x_2, y_2), dan (x_3, y_3), polinom yang menginterpolasi keempat titik tersebut adalah $p_3(x) = a_0 + a_1x + a_2x^2 + a_3x^3$, demikian seterusnya. Dengan cara yang sama kita dapat membuat polinom interpolasi berderajat n untuk n yang lebih tinggi asalkan tersedia $(n+1)$ buah titik data. Dengan menyulihkan (x_i, y_i) ke dalam persamaan polinom $p_n(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n$ untuk $i = 0, 1, 2, \ldots, n$, akan diperoleh n buah sistem persamaan lanjar dalam $a_0, a_1, a_2, \ldots, a_n$,

$$
\begin{aligned}
a_0 + a_1x_0 + a_2x_0^2 + \ldots + a_n\, x_0^n &= y_0 \\
a_0 + a_1x_1 + a_2x_1^2 + \ldots + a_n\, x_1^n &= y_1 \\
\ldots \qquad\qquad & \ldots \\
a_0 + a_1x_n + a_2x_n^2 + \ldots + a_n\, x_n^n &= y_n
\end{aligned}
$$

Solusi sistem persamaan lanjar ini, yaitu nilai $a_0, a_1, \ldots, a_n$, diperoleh dengan menggunakan metode eliminasi Gauss yang sudah anda pelajari. Sebagai contoh, misalkan diberikan tiga buah titik yaitu (8.0, 2.0794), (9.0, 2.1972), dan (9.5, 2.2513). Tentukan polinom interpolasi kuadratik lalu estimasi nilai fungsi pada $x = 9.2$. Polinom kuadratik berbentuk $p_2(x) = a_0 + a_1x + a_2x^2$. Dengan menyulihkan ketiga buah titik data ke dalam polinom tersebut, diperoleh sisten persamaan lanjar yang terbentuk adalah

$$a_0 + 8.0a_1 + 64.00a_2 = 2.0794$$

$$a_0 + 9.0a_1 + 81.00a_2 = 2.1972$$

$$a_0 + 9.5a_1 + 90.25a_2 = 2.2513$$

Penyelesaian sistem persamaan dengan metode eliminasi Gauss menghasilkan a_0 = 0.6762, a_1 = 0.2266, dan a_2 = -0.0064. Polinom interpolasi yang melalui ketiga buah titik tersebut adalah $p_2(x) = 0.6762 + 0.2266x - 0.0064x^2$. Dengan menggunakan polinom ini, maka nilai fungsi pada x = 9.2 dapat ditaksir sebagai berikut: $p_2(9.2) = 0.6762 + 0.2266(9.2) - 0.0064(9.2)^2 = 2.2192$.

III. Regresi Linier Berganda

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

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

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

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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

BAB II – DASAR TEORI

I. Eliminasi Gauss

Eliminasi Gauss adalah algoritme yang digunakan untuk menyelesaikan sistem persamaan linear. Metode ini dinamai dari matematikawan Carl Friedrich Gauss (1777–1855), walaupun metode ini sudah dikenal oleh matematikawan Tionghoa semenjak tahun 179 M.

Terdapat tiga jenis operasi baris elementer (OBE) yang dapat dilakukan:

1. Mengganti urutan dua baris
2. Mengalikan baris dengan angka yang bukan nol
3. Menambah suatu baris dengan baris yang lainnya

Dan langkah-langkah mendapat solusi SPL:

1. Nyatakan SPL dalam bentuk matriks augmented
2. Terapkan OBE pada matriks augmented sampai terbentuk matriks eselon baris
3. Pecahkan persamaan yang berkoresponden pada matriks eselon baris dengan teknik penyulihan mundur (*backward substitution*)

II. Eliminasi Gauss-Jordan

Eliminasi Gauss-Jordan merupakan pengembangan metode eliminasi Gauss. Operasi baris elementer (OBE) yang dapat dilakukan sama dengan eliminasi Gauss, dan langkah-langkah mendapat solusi SPL:

1. Nyatakan SPL dalam bentuk matriks augmented
2. Terapkan OBE pada matriks augmented sampai terbentuk matriks eselon baris tereduksi
3. Tidak diperlukan lagi substitusi secara mundur untuk memperoleh nilai-nilai variabel. Nilai variabel langsung diperoleh dari matriks augmented akhir.

III. Determinan

Dalam bidang aljabar linear, determinan adalah nilai yang dapat dihitung dari unsur suatu matriks persegi. Determinan matriks *A* ditulis dengan tanda det(*A*), det *A*, atau |*A*|. Determinan dapat dianggap sebagai faktor penskalaan transformasi yang digambarkan oleh matriks.

Apabila matriksnya berbentuk 2 × 2, rumus untuk mencari determinan adalah:

$$|\mathrm{A}| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = \mathrm{ad} - \mathrm{bc}$$

Apabila matriksnya berbentuk 3 × 3 matrix *A*, rumusnya adalah:

$$|A| = \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = aei + bfg + cdh - ceg - bdi - afh$$

Untuk matriks segitiga atas (upper triangular): semua elemen di bawah diagonal utama adalah nol

$$|A| = \begin{vmatrix} a & b & c \\ 0 & e & f \\ 0 & 0 & i \end{vmatrix} = aei$$

Untuk matriks segitiga bawah (lower triangular): semua elemen di atas diagonal utama adalah nol

$$|A| = \begin{vmatrix} a & 0 & 0 \\ d & e & 0 \\ g & h & i \end{vmatrix} = aei$$

Secara umum, untuk matriks segitiga A berukuran n x n, $|A| = a_{11}, a_{22}, a_{33}, ..., a_{nn}$

Determinan matriks A dapat diperoleh dengan melakukan OBE pada matriks A sampai diperoleh matriks segitiga (segitiga bawah atau atas).

Misalkan A adalah matriks n x n. Matriks B adalah matriks yang diperoleh dengan memanipulasi matriks A. Determinan B:

• B adalah perkalian sebuah baris A dengan k, maka det(B) = k det(A)
• B adalah pertukaran dua baris A, maka det(B) = –det(A)
• B adalah sebuah baris A ditambah k kali baris lain A, maka det(B) = det(A)

III. Balikan

Misalkan A adalah matriks persegi berukuran n x n. Balikan (inverse) matriks A adalah A^{-1} sedemikian sehingga $AA^{-1} = A^{-1}A = I$. Metode eliminasi Gauss-Jordan dapat digunakan untuk menghitung matriks balikan. Sebuah matriks hanya mempunyai balikan jika dan hanya jika $|A| \neq 0$.

Untuk matriks A yang berukuran n x n, matriks balikannya, yaitu A^{-1} dicari dengan cara berikut:

$[A|I] \sim [I|A^{-1}]$

yang dalam hal ini I adalah matriks identitas berukuran n x n. Metode eliminasi Gauss-Jordan diterapkan secara simultan untuk A maupun I.

Pada SPL Ax = b, solusi SPL adalah $x = A^{-1} b$. Jika A tidak mempunyai balikan, maka Ax = b tidak memiliki solusi yang tunggal (unik). Namun, jika A mempunyai balikan, maka SPL Ax = b memiliki solusi unik.

Pada SPL homogen Ax = 0, SPL hanya memiliki solusi trivial jika A memiliki balikan. Jika A tidak memiliki balikan, maka SPL memiliki solusi non-trivial.

IV. Matriks Kofaktor

Misalkan A adalah matriks berukuran n x n. Didefinisikan:

M_{ij} = minor entri a_{ij}

= determinan upa-matriks (submatrix) yang elemen-elemennya tidak berada pada baris i dan kolom j

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

Dengan menggunakan kofaktor, maka determinan matriks dapat dihitung dengan salah satu dari persamaan berikut:

$\det(A) = a_{11}C_{11} + a_{12}C_{12} + \ldots + a_{1n}C_{1n}$

$\det(A) = a_{21}C_{21} + a_{22}C_{22} + \ldots + a_{2n}C_{2n}$

$\vdots$

$\det(A) = a_{n1}C_{n1} + a_{n2}C_{n2} + \ldots + a_{nn}C_{nn}$

$\det(A) = a_{11}C_{11} + a_{21}C_{21} + \ldots + a_{n1}C_{n1}$

$\det(A) = a_{12}C_{12} + a_{22}C_{22} + \ldots + a_{n2}C_{n2}$

$\vdots$

$\det(A) = a_{1n}C_{1n} + a_{2n}C_{2n} + \ldots + a_{nn}C_{nn}$

V. Matriks Adjoin

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

$$\begin{bmatrix} C11 & \cdots & C1n \\ \vdots & \ddots & \vdots \\ Cn1 & \cdots & Cnn \end{bmatrix}$$

Adjoin dari A adalah transpose matriks kofaktor. Balikan matriks A dapat dihitung dengan menggunakan rumus:

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

VI. Kaidah Cramer

Jika Ax = b adalah SPL yang terdiri dari n persamaan linier dengan n peubah (variable) sedemikian sehingga det(A) □ 0, maka SPL tersebut memiliki solusi yang unik yaitu:

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

yang dalam hal ini, A_j adalah matriks yang diperoleh dengan mengganti entri pada kolom ke-j dari A dengan entri dari matriks:

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

VII. Interpolasi

Diberikan *n+1* buah titik berbeda, $(x_0, y_0),(x_1, y_1), \ldots, (x_n, y_n)$. Polinom $p_n(x)$ melalui semua titik-titik tersebut sedemikian sehingga

$y_i = p_n(x_i)$ untuk $i = 0, 1, 2, \ldots, n$

$p_n(x)$ dapat digunakan untuk menghitung perkiraan nilai y di $x = a$, yaitu $y = p_n(a)$.

Polinom interpolasi derajat n yang melalui titik-titik $(x_0, y_0),(x_1, y_1), \ldots, (x_n, y_n)$ adalah $p_n(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$

Polinom interpolasi berderajat n untuk n yang lebih tinggi:

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

asalkan tersedia $(n+1)$ buah titik data.

Dengan menyulihkan (x_i, y_i) ke dalam persamaan polinom di atas $y = p_n(x)$ untuk $i = 0, 1, 2, \ldots, n$, akan diperoleh $n + 1$ buah persamaan lanjar dalam $a_0, a_1, a_2, \ldots, a_n$,

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

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

$a_0 + a_1 x_2 + a_2 x_2^2 + \ldots + a_n x_2^n = y_2$

... ...

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

Solusi sistem persamaan lanjar ini diperoleh dengan menggunakan metode eliminasi Gauss

VII. Regresi Linier Berganda

Regresi Linear Berganda adalah model regresi linear dengan melibatkan lebih dari satu variable bebas. Model regresi linear berganda ditentukan sbb:

$$h_w(x) = w_0 + w_1x_1 + w_2x_2 + \cdots + w_mx_m$$

$$h_w(x) = w_0 + \sum_{i=0}^{m} w_ix_i$$

Dimana w merupakan nilai yang akan dicari sedemikian sehingga nilai w menjadi optimal dan x merupakan variable bebas atau input. Proses pencarian nilai w dapat dilakukan dengan menggunakan pendekatan *least square*, *maximum likehood*, atau algoritme *gradient descent*. Pada dasarnya, pencarian nilai w dilakukan hingga nilai error yang didapatkan dari fungsi error merupakan nilai yang paling minimal. Fungsi error yang digunakan sebagai berikut:

$$E(w) = \frac{1}{2N}\sum_{i=1}^{N}(h_w(x^i) - t^i)^2$$

Dimana N merupakan banyaknya data input, $h_w(x^i)$ merupakan model regresi linear, dan t adalah target output yang seharusnya.

BAB III – IMPLEMENTASI PROGRAM

Program terdiri dari 2 tipe data bentukan utama dan 5 fungsi utama yang tersusun atas beberapa fungsi-fungsi pendukung.

Tipe data bentukan, antara lain:

1. Matriks

Matriks berfungsi untuk membuat matriks (array of array/ 2D array), serta menyediakan fungsi-fungsi yang dapat dilakukan oleh matriks pada umumnya. Matriks memiliki 4 atribut, yaitu Mat, nBrs, nKol, dan idxMin. Mat adalah container dari matriks yang menyimpan nilai dari matriks, nBrs adalah jumlah baris dalam matriks, nKol adalah jumlah kolom, dan idxMin adalah indeks pertama dari suatu matriks.

Berikut adalah method yang tersedia pada Matriks:

a) Kostruktor

Membentuk matriks baru dan mendefinisikan nilai dari nBrs dan nKol serta ukuran dari Mat.

i. Matriks (int dimensi)

Membentuk matriks persegi berukuran dimensi x dimensi, serta memberi nilai dimensi kepada nBrs dan nKol.

ii. Matriks (int baris, int kolom)

Membentuk matriks persegi berukuran baris x kolom, serta memberi nilai baris kepada nBrs sedangkan nilai kolom kepada nKol.

iii. Matriks (float[][] tabel)

Menerima elemen-elemen berbentuk 2D array dan memberikannya kepada atribut Mat. nBrs adalah panjang dari array utama, sedangkan nKol adalah panjang dari array yang merupakan elemen dari array utama.

iv. Matriks (Matriks M)

Menyalin matriks lain dan mengubah nilai atribut menjadi sama dengan atribut matriks lain.

v. Matriks ()

Menerima inputan berupa nama file yang berformat .txt dan mengambil matriks yang tersedia pada file tersebut. nBrs dan nKol sesuai dengan panjang baris dan panjang kolom dari matriks yang diambil.

b) Getter and Setter

i. get (int i, int j)

Mendapatkan nilai dari matriks pada baris i dan kolom j.

ii. set (int i, int j, float val)

Mengubah nilai matriks pada baris i dan kolom j menjadi val.

iii. getnBrs ()
Mendapatkan nilai nBrs pada suatu matriks.
iv. getnKol ()
Mendapatkan nilai nKol pada suatu matriks.
v. getIdxMin()
Mendapatkan nilai idxMin.

c) Baca dan Tulis Matriks
i. bacaMatriks()
Memberikan nilai elemen pada atribut Mat dari suatu matriks.
ii. bacaMatriksSPL()
Memberikan nilai elemen pada atribut Mat dari suatu matriks untuk pengerjaan SPL dan nantinya akan diubah ke bentuk MAugmented.
iii. toString()
Mengubah matriks ke bentuk string untuk mempermudah pembuatan file keluaran.

d) Method fungsi untuk tugas
i. transpose ()
Menghasilkan matriks baru yang merupakan transpose dari matriks awal. Matriks dikatakan transpose apabila nKol transpose = nBrs dan nBrs transpose = nKol. Sedangkan elemen Transpose[i,j]=Matriks[j,i].
ii. kaliMatriks (Matriks m2)
Mengalikan matriks berukuran i x j dan k x l dan menghasilkan matriks berukuran i x l.
iii. kaliKonstanta (double c)
Mengalikan semua elemen Mat dengan konstanta c.
iv. gaussM ()
Menghasilkan matriks baru yang merupakan representasi dari hasil penerapan eliminasi gauss. Matriks hasil berupa matriks segitiga bawah dengan elemen diagonal utamanya adalah 1 atau 0.
v. gaussjorM ()
Menghasilkan matriks baru yang merupakan representasi dari hasil penerapan eliminasi gauss-jordan. Matriks hasil berupa matriks segitiga atas dan bawah dengan elemen diagonal utamanya adalah 1 atau 0.
vi. determinan ()
I.S Matriks terdefinisi nilainya
F.S Apabila matriks tidak persegi, mengeluarkan “Matriks tidak persegi”.
Apabila persegi, menghitung determinan dengan menggunakan ekspansi kofaktor.

vii. detred ()

I.S Matriks terdefinisi nilainya
F.S Apabila matriks tidak persegi, mengeluarkan "Matriks tidak persegi". Apabila persegi, menghitung determinan dengan menggunakan reduksi baris.

viii. makeMinor(int k, int l)
Membuat minor dari elemen matriks pada baris k, kolom l.
ix. invers ()
Mengembalikan matriks berupa invers dari suatu matriks. Perhitungan invers menggunakan adjoin.
x. InversOBE ()
Mengembalikan matriks berupa invers dari suatu matriks. Perhitungan invers mengacu pada A.I=I.A-1 dan menggunakan OBE.

2. MAugmented

MAugmented adalah tipe data bentukan yang merujuk pada matriks sebagai dasarnya. MAugmented membuat dua matriks berbeda, yaitu MatKoef dan MatVal. MatKoef adalah matriks yang menyimpan nilai dari koefisien persamaan, sedangkan MatVal adalah matriks yang menyimpan nilai dari hasil persamaan tersebut.

Berikut adalah method yang terdapat pada MAugmented:

a) Konstruktor

i. MAugmented ()
Membaca matriks pada file dan menyalinnya kedalam program sebagai MAugmented dengan MatKoef berukuran m x (n-1) dan MatVal berukuran m x 1.
ii. MAugmented (int i, int j)
Membuat MAugmented dengan MatKoef berukuran i x (j-1) dan MatVal berukuran i x 1.
iii. MAugmented (float[][] mat)
Menerima elemen-elemen berbentuk 2D array dan menyalinnya menjadi elemen MatKoef dan MatVal. nBrs adalah panjang dari array utama, sedangkan nKol adalah panjang dari array yang merupakan elemen dari array. Ukuran MatKoef adalah nBrs x (nKol-1), sedangkan ukuran MatVal adalah nBrs x 1.
iv. MAugmented (Matriks M)
Menyalin matriks lain menjadi MAugmented. MatKoef berukuran M.nBrs x (M.nKol -1) dan MatVal berukuran M.nBrs x 1.
v. makeMatKoef ()
Fungsi antara untuk membantu pembuatan MatKoef
vi. makeMatVal ()
Fungsi antara untuk membantu pembuatan MatVal.

b) Getter and Setter

i. isBarisKoefNol (int i)

Mengembalikan true apabila pada baris ke i semua elemen MatKoefnya bernilai 0. Jika tidak, mengembalikan false.

ii. isInconsistent ()

Mengembalikan nilai true apabila terdapat baris MatKoef yang hanya berelemen 0, namun elemen pada MatVal dengan baris yang sama tidak bernilai 0.

iii. nBrsEff ()

Mengembalikan jumlah baris efektif . Baris efektif adalah ketika elemen MatKoef dan MatVal pada baris yang sama tidak hanya bernilai 0.

c) Baca dan Tulis

i. setMatKoef ()

Memberi nilai pada elemen matriks MatKoef.

ii. setMatVal ()

Memberi nilai pada elemen matriks MatVal.

iii. kalimatSolusi ()

Menuliskan solusi dari SPL dalam bentuk string.

d) Soal

i. inversSPL()

Menentukan solusi dari suatu SPL menggunakan invers matriks x=A-1b.

ii. gauss ()

Mengembalikan MAugmented berupa MatKoef yang merupakan matriks segitiga bawah dengan elemen diagonal utama 1 menggunakan OBE. Nilai elemen MatVal juga berubah dikarenakan OBE.

iii. gaussjor ()

Mengembalikan MAugmented berupa MatKoef yang merupakan matriks segitiga atas dan bawah dengan elemen diagonal utama 1 menggunakan OBE. Nilai elemen MatVal juga berubah dikarenakan OBE.

iv. cramer ()

Menentukan solusi suatu SPL menggunakan metode Cramer, yaitu dengan membagi determinan matriks mula-mula dengan determinan matriks hasil substitusi MatVal ke MatKoef.

3. Regresi

Regresi terdiri dari Matriks tabEQ dan tabTaksir serta MAugmented tabNormal dan solusiTabNormal. tabEQ adalah matriks yang barisnya merepresentasikan banyaknya persamaan dan kolomnya merepresentasikan banyaknya variabel dalam persamaan tersebut. tabTaksir adalah matriks untuk menyimpan persamaan yang ingin ditaksir, namun kolom terakhir untuk setiap baris akan digunakan untuk menyimpan nilai taksiran. tabNormal adalah MAugmented untuk normal equation pada regresi berganda. solusiTabNormal adalah MAugmented hasil penerapan gauss-jordan pada MAugmented tabNormal.

a. Getter and setter

i. getTabEQ ()

Mengembalikan matriks tabEQ.

ii. getSolusiTabNormal ()

Mengembalikan MAugmented solusiTabNormal

iii. getTabTaksir ()

Mengembalikan matriks tabTaksir.

iv. getTabNormal ()

Mengembalikan MAugmented tabNormal.

v. getArrayB ()

Membuat array yang berelemen konstanta dari tiap baris

b. Baca dan Tulis

i. makeTabTaksir (int nTaksir, int k)

Menginisialisasi tabTaksir dengan ukuran nTaksir x (k+1) dan memberi nilai untuk elemen tabTaksir.

ii. makeTabTaksirFile (Matriks temp)

Menginisialisasi tabTaksir dengan ukuran temp.nBrs x (temp.nKol +1) dan mengisi elemennya sesuai dengan file yang terbaca pada matriks temp.

iii. makeTabEQ (int n, int k)

Menginisialisasi tabTaksir dengan ukuran n x k dan memberi nilai untuk elemen tabEQ.

iv. makeTabNormal (int l)

Menginisialisasi tabTaksir dengan ukuran (l+1) x (l+2) dan memberi nilai untuk elemen tabNormal.

v. taksiranToString ()

Menghasilkan hasil taksiran dalam bentuk string.

c. Soal

i. Regresi (boolean inputFile)

Program utama yang menjalankan fitur regresi pada matriks.

ii. taksirNilai ()

Menaksir nilai dan menaruh hasil tersebut di tabTaksir pada kolom terakhir di setiap baris dengan persamaan yang relevan.

4. Interpolasi

Terdiri dari 2 class, yaitu Point dan Interpolasi. Point merupakan tipe data bentukan untuk menyimpan hasil masukan dari user yang berupa poin pada bidang kartesian. Interpolasi memiliki 7 atribut, yaitu arrP, arr, M, Mfile, x, y, sumMultKoef. arrP adalah sebuah array of Point yang digunakan untuk menyimpan hasil masukan user. arr adalah array of float yang digunakan untuk menyimpan hasil persamaan P(x) setelah melakukan eliminasi metode gauss. M adalah matriks tempat menyimpan persamaan berdasarkan point yang dimasukan user. Matriks Mfile merupakan tempat menyimpan matriks apabila masukan berasal dari suatu file.

5. MakeFile

Class yang digunakan seperti prosedur untuk membuat file .txt dari string yang dimasukkan ke parameter konstruktornya.

6. Menu

Class yang menghasilkan interface command line ketika dibuat instancenya. Interface dibuat dalam bentuk class agar lebih modular dan mudah dibaca.
Method:
Menu()
Konstruktor yang menghasilkan menu utama ketika dipanggil. Meminta input dan memasukkan input tersebut dalam switch statement yang memanggil menu selanjutnya.
menuSPL()
Fungsinya hampir sama dengan Menu(), meminta input dan memasukkan dalam switch statement untuk memanggil fungsi run selanjutnya.
runGauss(), runGaussJor(), runInversSPL(), runCramer()
Prosedur untuk menjalankan penyelesaian SPL sesuai dengan pilihan. Skema fungsi-fungsi ini hampir sama, dengan awalnya menentukan cara membaca matriks, dan di akhir menentukan menyimpan hasil ke file atau tidak.
Perbedaan tiap prosedur ini terdapat di cara pembuatan string result, yang menyesuaikan dengan pilihan cara penyelesaian SPL pengguna. Misalnya apabila dipilih Gauss atau GaussJor, matriks yang dibuat dari pembacaan akan berupa MAugmented.
menuDet(), menuInvers()
Serupa dengan prosedur run untuk SPL, tetapi diawali dengan memilih cara penyelesaian determinan atau invers. Setelah ditentukan cara penyelesaian, prosedur pembentukan Martiks dan string Result hampir sama dengan run SPL.
menuRegresi(), Interpolasi()
Serupa dengan run SPL, diawali dengan memilih cara pembentukan matriks dan membuat string result, kemudian memilih untuk disimpan dalam .txt atau tidak.

BAB IV – IMPLEMENTASI PROGRAM

Soal 1

Uji kasus 1.a., 1.b., dan 1.c. berjalan dengan lancar.

```
Baca file (tanpa ext.): 1a
matriks:
1.00    1.00    -1.00   -1.00   1.00
2.00    5.00    -7.00   -5.00   -2.00
2.00    -1.00   1.00    3.00    4.00
5.00    2.00    -4.00   2.00    6.00

hasil gaussjor:
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

variabel x:
Tidak ada solusi
```

```
matriks:
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

hasil gaussjor:
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

variabel x:
x1 = 3.00 + 1.00x5
x2 = 0.00 + 2.00x5
x4 = -1.00 + 1.00x5
x3, x5 bilangan riil
```

```
matriks:
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

hasil gaussjor:
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

variabel x:
x2 = 1.00 - 1.00x6
x4 = -2.00 - 1.00x6
x5 = 1.00 + 1.00x6
x1, x3, x6 bilangan riil
```

Untuk uji kasus 1.d., masih belum tepat. Nilai yang kami dapat berbeda beberapa desimal dari yang seharusnya, seperti yang ditunjukkan berikut (nilai yang kami dapatkan di program kami ada di sebelah kanan, nilai yang kami dapatkan di kalkulator ada di sebelah kiri):

	variabel x:
x_1 = 36.665857039901949855	x1 = 37.00
x_2 = -648.49821194062181386	x2 = -657.05
x_3 = 3483.0393116798341714	x3 = 3536.57
x_4 = -7876.2666953873619394	x4 = -8007.26
x_5 = 7906.0417762996068845	x5 = 8043.61
x_6 = -2907.4290904463319101	x6 = -2959.44

Kami duga, itu karena matriks kami tidak dapat menampung variable yang lebih dari dua desimal di belakang koma. Berikut adalah output lengkapnya:

```
matriks:
1.00    0.50    0.33    0.25    0.20    0.17    1.00
0.50    0.33    0.25    0.20    0.17    0.14    0.00
0.33    0.25    0.20    0.17    0.14    0.13    0.00
0.25    0.20    0.17    0.14    0.13    0.11    0.00
0.20    0.17    0.14    0.13    0.11    0.10    0.00
0.17    0.14    0.13    0.11    0.10    0.09    0.00

hasil gaussjor:
1.00    0.00    0.00    0.00    0.00    0.00    37.00
0.00    1.00    0.00    0.00    0.00    0.00    -657.05
0.00    0.00    1.00    0.00    0.00    0.00    3536.57
0.00    0.00    0.00    1.00    0.00    0.00    -8007.26
0.00    0.00    0.00    0.00    1.00    0.00    8043.61
0.00    0.00    0.00    0.00    0.00    1.00    -2959.44

variabel x:
x1 = 37.00
x2 = -657.05
x3 = 3536.57
x4 = -8007.26
x5 = 8043.61
x6 = -2959.44
```

Soal 2

Uji kasus 2.a. dan 2.b. berjalan dengan lancar.

```
matriks:
1.00    -1.00   2.00    -1.00   -1.00
2.00    1.00    -2.00   -2.00   -2.00
-1.00   2.00    -4.00   1.00    1.00
3.00    0.00    0.00    -3.00   -3.00

hasil gaussjor:
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

variabel x:
x1 = -1.00 + 1.00x4
x2 = 0.00 + 2.00x3
x3, x4 bilangan riil
```

```
matriks:
2.00    0.00    8.00    0.00    8.00
0.00    1.00    0.00    4.00    6.00
-4.00   0.00    6.00    0.00    6.00
0.00    -2.00   0.00    3.00    -1.00
2.00    0.00    -4.00   0.00    -4.00
0.00    1.00    0.00    -2.00   0.00

hasil gaussjor:
1.00    0.00    0.00    0.00    0.00
0.00    1.00    0.00    0.00    2.00
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

variabel x:
x1 = 0.00
x2 = 2.00
x3 = 1.00
x4 = 1.00
```

Soal 3

Uji kasus 3.a. dan 3.b. berjalan dengan lancar.

```
matriks:
8.00    1.00    3.00    2.00    0.00
2.00    9.00    -1.00   -2.00   1.00
1.00    3.00    2.00    -1.00   2.00
1.00    0.00    6.00    4.00    3.00

hasil gaussjor:
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

variabel x:
x1 = -0.22
x2 = 0.18
x3 = 0.71
x4 = -0.26
```

```
matriks:
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    0.25    0.91    0.25    0.00    0.25    0.91    16.13
0.04    0.00    0.00    0.75    0.04    0.00    0.61    0.75    0.04    7.04

hasil gaussjor:
1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    3.00
0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    -17.00
0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    21.50
0.00    0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    18.50
0.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00    0.00    5.01
0.00    0.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00    -8.50
0.00    0.00    0.00    0.00    0.00    0.00    1.00    0.00    0.00    -15.50
0.00    0.00    0.00    0.00    0.00    0.00    0.00    1.00    0.00    23.50
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    1.00    5.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    15.50
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00

variabel x:
Tidak ada solusi
```

Soal 4

Uji kasus 4 berjalan dengan lancar.

```
1.00    1.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    -1.00   0.00    1.00    -1.00   0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    -1.00   0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    1.00    -1.00   0.00    0.00    0.00    0.00    0.00
0.00    0.00    10.00   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    5.00    0.00    1.00    -1.00   0.00    0.00
0.00    0.00    0.00    20.00   0.00    0.00    0.00    0.00    0.00    1.00    0.00
5.00    0.00    0.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00    200.00
0.00    0.00    0.00    0.00    15.00   0.00    0.00    0.00    1.00    -1.00   0.00
0.00    10.00   0.00    0.00    0.00    0.00    1.00    0.00    0.00    -1.00   0.00

hasil gaussjor:
1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    6.67
0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    -3.33
0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    -3.33
0.00    0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    0.00    -6.67
0.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00    0.00    0.00    -3.33
0.00    0.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00    0.00    -3.33
0.00    0.00    0.00    0.00    0.00    0.00    1.00    0.00    0.00    0.00    166.67
0.00    0.00    0.00    0.00    0.00    0.00    0.00    1.00    0.00    0.00    200.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    1.00    0.00    183.33
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    1.00    133.33

variabel x:
x1 = 6.67
x2 = -3.33
x3 = -3.33
x4 = -6.67
x5 = -3.33
x6 = -3.33
x7 = 166.67
x8 = 200.00
x9 = 183.33
x10 = 133.33
```

Soal 5

Uji kasus 5 berjalan dengan sedikit kendala, output masih dalam bentuk notasi saintifik, belum diubah ke angka desimal. Namun hasil taksiran sudah benar, diuji dengan substitusi nilai x ke persamaan.

```
P(x)=-0.0229764+0.23999691x+0.19741522x^2-5.5465847E-5x^3+0.026120555x^4-5.393747E-5x^5+1.4089713E-5x^6
Masukkan nilai yang ingin ditaksir: 0.2

Nilai P(0.2)=0.03296092
```

```
P(x)=-0.0229764 + 0.23999691x + 0.19741522x^2-5.5465847E-5x^3 + 0.026120555x^4-5.393747E-5x^5 + 1.4089713E-5x^6
Masukkan nilai yang ingin ditaksir: 0.55

Nilai P(0.55)=0.17111865
```

```
P(x)=-0.0229764 + 0.23999691x + 0.19741522x^2-5.5465847E-5x^3 + 0.026120555x^4-5.393747E-5x^5 + 1.4089713E-5x^6
Masukkan nilai yang ingin ditaksir: 0.85

Nilai P(0.85)=0.3372359
```

```
P(x)=-0.0229764 + 0.23999691x + 0.19741522x^2-5.5465847E-5x^3 + 0.026120555x^4-5.393747E-5x^5 + 1.4089713E-5x^6
Masukkan nilai yang ingin ditaksir: 1.28

Nilai P(1.28)=0.67754185
```

Soal 6

Persamaan polinomial:

Tidak sesuai dengan persamaan yang didapatkan dari website wolfram. Penyebabnya adalah nilai x yang butuh presisi yang besar dan nilai x selalu dipangkatkan hingga pangkat 9. Karena keterbatasan kemampuan, maka presisi yang dibutuhkan menurun dan menyebabkan kekeliruan tersebut.

Soal 1:

```
P(x)=1.03265536E9-1.35838771E9x+7.8476211E8x^2-2.61230976E8x^3+5.5200864E7x^4-7677552.0x^5+702875.5x^6-40858.254x^7+1369.5857x^8-20.199516x^9
Masukkan nilai yang ingin ditaksir: 5.806

Nilai P(5.806)=27104.0
```

Soal 2:

```
P(x)=1.03265536E9-1.35838771E9x+7.8476211E8x^2-2.61230976E8x^3+5.5200864E7x^4-7677552.0x^5+702875.5x^6-40858.254x^7+1369.5857x^8-20.199516x^9
Masukkan nilai yang ingin ditaksir: 8.967

Nilai P(8.967)=172544.0
```

Soal 3

```
P(x)=1.03265536E9-1.35838771E9x+7.8476211E8x^2-2.61230976E8x^3+5.5200864E7x^4-7677552.0x^5+702875.5x^6-40858.254x^7+1369.5857x^8-20.199516x^9
Masukkan nilai yang ingin ditaksir: 9.5

Nilai P(9.5)=3072.0
```

Titik uji tidak valid karena diluar dari rentang [x0,xn]

Soal 4

```
P(x)=1.03265536E9-1.35838771E9x+7.8476211E8x^2-2.61230976E8x^3+5.5200864E7x^4-7677552.0x^5+702875.5x^6-40858.254x^7+1369.5857x^8-20.199516x^9
Masukkan nilai yang ingin ditaksir: 2.724

Nilai P(2.724)=8827001.0
```

Titik uji tidak valid karena diluar dari rentang [x0,xn]

Soal 7

```
7.txt
0.2 0.3428
0.4 0.4189
0.6 0.4684
0.8 0.5072
1.0 0.5379
1.2 0.5609
1.4 0.5762
1.6 0.5837
1.8 0.5837
2.0 0.5767
```

```
Baca file (tanpa ext.): 7

P(x)=0.17280385 + 1.3425181x-3.5194025x^2 + 6.712619x^3-8.381457x^4 + 6.86952x^5-3.6796098x^6
 + 1.2385683x^7-0.23740023x^8 + 0.019740092x^9
Masukkan nilai yang ingin ditaksir: 2.4

Nilai P(2.4)=0.5477028
```

Dilakukan interpolasi sesuai dengan testcase 7 yang disediakan di spesifikasi tugas besar dengan n = 10, sehingga jarak antar titik sebesar 0.2. Didapatkan polinom derajat 10 yang digunakan untuk mentaksir x = 2.4, didapatkan P(2.4) = 0.5477028, yang mendekati nilai asli dari f(2.4) = 0.5445.

Soal 8

```
MINGW64:/c/Users/aditya/Documents/New folder/ngoding/Java/Algeo01-19064
matriks persamaan awal:
72.40   76.30   29.18   0.90
41.60   70.30   29.35   0.91
34.30   77.10   29.24   0.96
35.10   68.00   29.27   0.89
10.70   79.00   29.78   1.00
12.90   67.40   29.39   1.10
8.30    66.80   29.69   1.15
20.10   76.90   29.48   1.03
72.20   77.70   29.09   0.77
24.00   67.70   29.60   1.07
23.20   76.80   29.38   1.07
47.40   86.60   29.35   0.94
31.50   76.90   29.63   1.10
10.60   86.30   29.56   1.10
11.20   86.00   29.48   1.10
73.30   76.30   29.40   0.91
75.40   77.90   29.28   0.87
96.60   78.70   29.29   0.78
107.40  86.80   29.03   0.82
54.90   70.90   29.37   0.95

matriks persamaan normal:
20.00   863.10  1530.40 587.84  19.42
863.10  54876.89        67000.09        25283.40        779.48
1530.40 67000.09        117912.32       44976.87        1483.44
587.84  25283.40        44976.87        17278.51        571.12

solusi persamaan normal:
1.00    0.00    0.00    0.00    -3.51
0.00    1.00    0.00    0.00    -0.00
0.00    0.00    1.00    0.00    0.00
0.00    0.00    0.00    1.00    0.15

konstanta B:
[-3.5113497, -0.0026244328, 7.9925323E-4, 0.1542749]

hasil taksiran:
taksiran:
x1      x2      x3      y
1.00    2.00    3.00    -3.05
2.00    2.00    2.00    -3.21
4.00    5.00    6.00    -2.59
```

Dilakukan percobaan regresi dengan input persamaan awal yang disediakan di spesifikasi tugas besar, dihasilkan persamaan normal sesuai dengan yang diharapkan. Kemudian persamaan normal tersebut diselesaikan dengan eliminasi Gauss-Jordan untuk mendapatkan solusi. Dari solusi persamaan normal tersebut didapatkan konstanta-konstanta B0 hingga Bn, yang kemudian dapat digunakan untuk melakukan taksiran nilai.

BAB VI – KESIMPULAN, SARAN, REFLEKSI

Hasil yang telah dicapai kelompok kami dalam tubes ini cukup memuaskan. Uji kasus yang kami lakukan kebanyakan sudah menghasilkan output yang diharapkan, walau masih terdapat beberapa bug. Bug-bug tersebut sebenarnya masih bisa diperbaiki, namun karena

Hal yang dapat kami kembangkan dari program ini, tentunya, adalah implementasi GUI. Dan kami sedikit kecewa dengan hasil program kami yang masih ada beberapa bug. Jika kami telah mengerjakan tugas ini dengan lebih efektif,seharusnya beberapa bug tersebut dapat teratasi.

Refleksi dari Pedro: Cukup susah bagi saya untuk mengerjakan tubes ini. Saya hanya familiar dengan operasi-operasi standar (if else, loop, fungsi), sementara kedua teman saya tampaknya tidak memiliki masalah apa-apa dalam menangani library, class, dan input-output file, di antara hal lain. Saya jadi merasa kurang bisa berkontribusi banyak.

Tapi saya jadi belajar manajemen waktu yang baik melalui tubes ini. Dan saya juga banyak menambah wawasan dari tubes ini.

Refleksi dari Leo: Sedikit sulit membagi waktu karena pertama kalinya mengerjakan tugas besar yang tenggat waktunya cukup sempit. Selain itu standar waktu antara pengerjaan tugas besar dasar pemrograman dan aljabar linier dan geometri ternyata berbeda jauh dan bebannya sangat terasa disini. Di saat-saat terakhir saya merasa panik karena tiba-tiba cukup banyak bug yang muncul saat mengerjakan studi kasus. Self reminder, sebaiknya studi kasus langsung dikerjakan saat selesai membuat suatu spek agar beban kerja bisa lebih terlihat dan jika ada bug bisa segera diatasi.

DAFTAR REFERENSI

1. Determinan. (2017, Desember 1). Di Wikipedia, Ensiklopedia Bebas. Diakses pada Oktober 1, 2020, dari https://id.wikipedia.org/wiki/Determinan

2. Regresi Linear. (2019, Februari 5). Di Wikipedia, Ensiklopedia Bebas. Diakses pada Oktober 1, 2020, dari https://id.wikipedia.org/wiki/Regresi_linear

3. Bremer, M., 2020. 18 - 36. [online] Mezeylab.cb.bscb.cornell.edu. Tersedia di: http://mezeylab.cb.bscb.cornell.edu/labmembers/documents/supplement%205%20-%20multiple%20regression.pdf [Diakses 2 Oktober 2020].

4. Slide bahan kuliah IF2123 tahun 2020