LAPORAN TUGAS BESAR 1 IF 2123 ALJABAR LINIER DAN GEOMETRI SISTEM PERSAMAAN LINIER, DETERMINAN DAN APLIKASINYA

Diajukan sebagai salah satu tugas mata kuliah Aljabar Linier dan Geometri pada Semester I Tahun Akademik 2020-2021

oleh

Muhammad Tito Prakasa	**13519007**
Fakhri Nail	**13519035**
Azmi Muhammad Syazwana	**13519151**

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2020

DAFTAR ISI

BAB 1

DESKRIPSI MASALAH

1.1 Abstraksi

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

$$
\begin{array}{c}
a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\
a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\
\vdots \qquad\qquad\qquad \vdots \\
a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n = b_m
\end{array}
$$

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

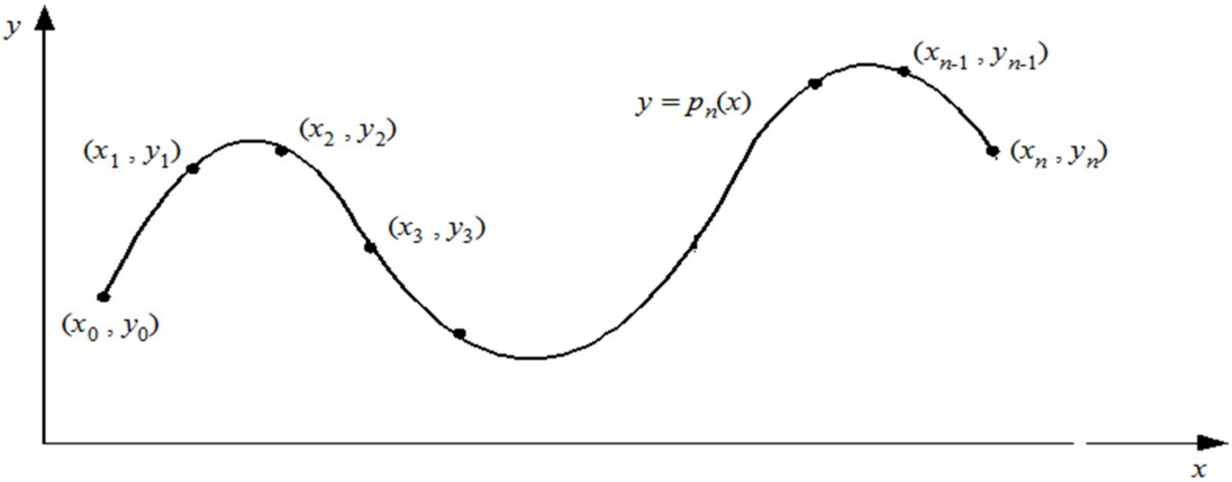
$$\det(M) = \begin{vmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{vmatrix}$$

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.

1.2 Interpolasi Polinom

Persoalan interpolasi polinom adalah sebagai berikut: Diberikan n+1 buah titik berbeda, (x0, y0), (x1, y1), ..., (xn, yn). Tentukan polinom pn(x) yang menginterpolasi (melewati) semua titik-titik tersebut sedemikian rupa sehingga yi = pn(xi) untuk i = 0, 1, 2, …, n.



Setelah polinom interpolasi pn(x) ditemukan, pn(x) dapat digunakan untuk menghitung perkiraan nilai y di sembarang titik di dalam selang [x0, xn].

Polinom interpolasi derajat n yang menginterpolasi titik-titik (x0, y0), (x1, y1), ..., (xn, yn). adalah berbentuk pn(x) = a0 + a1x + a2x^2 + … + anx^n. Jika hanya ada dua titik, (x0, y0) dan (x1, y1), maka polinom yang menginterpolasi kedua titik tersebut adalah p1(x) = a0 + a1x yaitu berupa persamaan garis lurus. Jika tersedia tiga titik, (x0, y0), (x1, y1), dan (x2, y2), maka polinom yang menginterpolasi ketiga titik tersebut adalah p2(x) = a0 + a1x + a2x^2 atau persaman kuadrat dan kurvanya berupa parabola. Jika tersedia empat titik, (x0, y0), (x1, y1), (x2, y2), dan (x3, y3), polinom yang menginterpolasi keempat titik tersebut adalah p3(x) = a0 + a1x + a2x^2 + a3x^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 (xi, yi) ke dalam persamaan polinom pn(x) = a0 + a1x + a2x^2 + … + anx^n untuk i = 0, 1, 2, …, n, akan diperoleh n buah sistem persamaan lanjar dalam a0, a1, $a2$, …, an,

$$a_0 + a_1x_0 + a_2x_0^2 + \ldots + a_nx_0^n = y_0$$

$$a_0 + a_1x_1 + a_2x_1^2 + \ldots + a_nx_1^n = y_1$$

$$\ldots \qquad \ldots$$

$$a_0 + a_1x_n + a_2x_n^2 + \ldots + a_nx_n^n = y_n$$

Solusi sistem persamaan lanjar ini, yaitu nilai $a_0, a_1, ..., a_n$, 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

$$\begin{aligned} a_0 + 8.0a_1 + 64.00a_2 &= 2.0794 \\ a_0 + 9.0a_1 + 81.00a_2 &= 2.1972 \\ a_0 + 9.5a_1 + 90.25a_2 &= 2.2513 \end{aligned}$$

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: $p_2(9.2) = 0.6762 + 0.2266(9.2) - 0.0064(9.2)^2 = 2.2192$.

1.3 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 βi dapat digunakan *Normal Estimation Equation for Multiple Linear Regression* sebagai berikut:

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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

BAB II

TEORI SINGKAT

2.1 Matriks *Augmented*

Matriks Augmented adalah matriks yang diperbesar dengan menambahkan kolom dari dua matriks tertentu atau kolom tambahan yang entrinya merupakan bilangan ruas kanan dari sistem persamaan. Sistem persamaan linear dapat dinyatakan secara ringkas dalam bentuk matriks *augmented*. Berikut merupakan contoh dari matriks *augmented.*

$$[A \mid \mathbf{b}] = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} & b_m \end{array}\right]$$

- Contoh:

$$\begin{aligned} x_1 + 3x_2 - 6x_3 &= 9 \\ 2x_1 - 6x_2 + 4x_3 &= 7 \\ 5x_1 + 2x_2 - 5x_3 &= -2 \end{aligned} \quad \longrightarrow \quad \begin{bmatrix} 1 & 3 & -6 & 9 \\ 2 & -6 & 4 & 7 \\ 5 & 2 & -5 & -2 \end{bmatrix}$$

2.2 Operasi Baris Elementer

Operasi baris elementer merupakan suatu operasi yang dapat diterapkan pada sebuah matriks. Ada tiga operasi baris elementer terhadap matriks *augmented*, yaitu:

1. Kalikan sebuah baris dengan konstanta tidak nol.
2. Pertukarkan dua buah baris.
3. Tambahkan sebuah baris dengan kelipatan baris lainnya.

2.3 Matriks Eselon Baris

Matriks eselon baris adalah matriks yang memiliki 1 utama pada setiap baris kecuali baris yang seluruhnya nol. Sifat-sifat matriks eselon baris yaitu:

1. Jika sebuah baris tidak terdiri dari nol seluruhnya, maka bilangan tidak nol pertama di dalam baris tersebut adalah 1 sehingga disebut 1 utama.
2. Jika ada baris yang seluruhnya nol, maka semua baris itu dikumpulkan pada bagian bawah matriks.

3. Di dalam dua baris berurutan yang tidak seluruhnya nol, maka 1 utama pada baris yang lebih rendah terdapat lebih jauh ke kanan daripada 1 utama pada baris yang lebih tinggi.

Berikut merupakan contoh matriks eselon baris.

$$\begin{bmatrix} 1 & * & * & * \\ 0 & 1 & * & * \\ 0 & 0 & 1 & * \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & * & * & * \\ 0 & 1 & * & * \\ 0 & 0 & 1 & * \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & * & * & * \\ 0 & 1 & * & * \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & * & * & * & * & * & * & * & * \\ 0 & 0 & 0 & 1 & * & * & * & * & * & * \\ 0 & 0 & 0 & 0 & 1 & * & * & * & * & * \\ 0 & 0 & 0 & 0 & 0 & 1 & * & * & * & * \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & * \end{bmatrix}$$

2.4 Matriks Eselon Baris Tereduksi

Matriks eselon baris tereduksi adalah matriks yang memiliki 1 utama pada setiap baris kecuali baris yang seluruhnya nol. Sifat matriks eselon baris tereduksi hampir sama seperti matriks eselon baris, tetapi perbedaannya yaitu pada matriks eselon baris tereduksi angka di atas 1 utama harus bernilai nol. Sifat-sifat matriks eselon baris yaitu:

1. Jika sebuah baris tidak terdiri dari nol seluruhnya, maka bilangan tidak nol pertama di dalam baris tersebut adalah 1 sehingga disebut 1 utama.
2. Jika ada baris yang seluruhnya nol, maka semua baris itu dikumpulkan pada bagian bawah matriks.
3. Di dalam dua baris berurutan yang tidak seluruhnya nol, maka 1 utama pada baris yang lebih rendah terdapat lebih jauh ke kanan daripada 1 utama pada baris yang lebih tinggi.
4. Setiap kolom yang memiliki 1 utama memiliki nol di tempat lain.

Berikut merupakan contoh matriks eselon baris.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & * \\ 0 & 1 & 0 & * \\ 0 & 0 & 1 & * \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 & * & * \\ 0 & 1 & * & * \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & * & 0 & 0 & 0 & * & * & 0 & * \\ 0 & 0 & 0 & 1 & 0 & 0 & * & * & 0 & * \\ 0 & 0 & 0 & 0 & 1 & 0 & * & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 1 & * & * & 0 & * \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & * \end{bmatrix}$$

2.5 Metode Eliminasi Gauss

Metode ini ditemukan oleh Carl Friedrich Gauss dan digunakan sebagai algoritme untuk menyelesaikan sistem persamaan linear. Metode ini merepresentasikan sistem persamaan linear menjadi bentuk matriks *augmented* lalu mengoperasikan nilai-nilai di dalam matriks melalu operasi baris elementer sehingga menjadi matriks eselon baris.

Langkah-langkah metode eliminasi Gauss untuk menyelesaikan sistem persamaan linear:

1. Nyatakan sistem persamaan linear tersebut dalam bentuk matriks augmented seperti yang telah dibahas pada poin 2.1 Matriks *Augmented*.
2. Menerapkan operasi baris elementer pada matriks augmented tersebut sampai menjadi matriks eselon baris. Operasi baris elementer yang dapat dilakukan ada 3 seperti yang telah dibahas di atas, yaitu:
   a. Mengalikan suatu baris dengan skalar tak nol.
   b. Menukar letak dua baris.
   c. Menambah suatu baris dengan kelipatan baris yang lain.

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_n \end{bmatrix} \sim\text{OBE}\sim \begin{bmatrix} 1 & * & * & \dots & * & * \\ 0 & 1 & * & \dots & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \vdots & 1 & * \end{bmatrix}$$

3. Setelah terbentuk matriks eselon baris, maka lakukan Teknik penyuluhan mundur (*backward substitution*) untuk mendapatkan nilai setiap variabel dalam sistem persamaan linear tersebut.

2.6 Metode Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan adalah salah satu metode untuk menyelesaikan sistem persamaan linear. Metode ini merupakan pengembangan dari metode eliminasi Gauss yang hasilnya menjadi lebih sederhana lagi. Selain itu, metode ini juga dapat digunakan untuk mencari invers dari sebuah matriks. Metode ini hampir sama seperti metode eliminasi Gauss, tetapi hasil dari metode ini bentuknya menjadi matriks eselon baris tereduksi bukan matriks eselon baris. Metode ini merepresentasikan sistem persamaan linear menjadi bentuk matriks *augmented* lalu mengoperasikan nilai-nilai di dalam matriks melalu operasi baris elementer sehingga menjadi matriks eselon baris tereduksi.

Langkah-langkah metode eliminasi Gauss-Jordan untuk menyelesaikan sistem persamaan linear:

1. Nyatakan sistem persamaan linear tersebut dalam bentuk matriks augmented seperti yang telah dibahas pada poin 2.1 Matriks *Augmented*.
2. Menerapkan operasi baris elementer pada matriks augmented tersebut sampai menjadi matriks eselon baris tereduksi. Operasi baris elementer yang dapat dilakukan ada 3 seperti yang telah dibahas di atas, yaitu:
   a. Mengalikan suatu baris dengan skalar tak nol.
   b. Menukar letak dua baris.
   c. Menambah suatu baris dengan kelipatan baris yang lain.

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} & b_m \end{bmatrix} \sim\text{OBE}\sim \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & * \\ 0 & 1 & 0 & \dots & 0 & * \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \vdots & 1 & * \end{bmatrix}$$

3. Setelah terbentuk matriks eselon baris tereduksi, tidak perlu lagi melakukan teknik penyuluhan mundur (*backward substitution*) untuk mendapatkan nilai setiap variabel dalam sistem persamaan linear tersebut karena nilai setiap variabelnya sudah langsung diperoleh dari matriks *augmented* akhir karena sudah berbentuk matriks eselon baris tereduksi.

Metode eliminasi Gauss-Jordan terdiri dari dua fase:

1. Fase Maju (*Forward Phase)* atau Fase Eliminasi Gauss
   - Menghasilkan nilai-nilai 0 di bawah 1 utama.
2. Fase Mundur (*Bacward Phase*)
   - Menghasilkan nilai-nilai 0 di atas 1 utama.

Cara mencari invers matriks menggunakan metode eliminasi gauss, yaitu dengan melakukan matriks *augmented* antara matriks yang ingin dicari inversnya dan matriks identitas (I). Setelah itu, ubah matriks yang ingin dicari inversenya menjadi matriks identitas dengan melakukan metode eliminasi Gauss-Jordan pada matriks *augmented* tersebut (secara simultan untuk A dan I).

2.7 Determinan

Determinan merupakan suatu nilai yang dapat dihitung dari unsur-unsur suatu matriks persegi. Matriks persegi sendiri adalah matriks yang memiliki jumlah baris dan kolom yang sama sehingga bentuknya terlihat seperti persegi. Determinan matriks A ditulis dengan tanda det(A), det A, atau |A|. Determinan dapat dianggap sebagai faktor penskalaan transformasi yang digambarkan oleh matriks.

Sifat-sifat determinan yaitu sebagai berikut:

- Jika matriks A dan B adalah matriks persegi yang berordo sama, maka
  det(AB) = det(BA) = det(A) x det(B)
- Jika A adalah matriks persegi dan A^T adalah transpose matriks A, maka
  det(A) = det(A^T)
- Jika A adalah matriks segitiga atas atau bawah, maka
  det(A) = a_{11} x a_{22} x … x a_{nn}

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix} \longrightarrow \det(A) = a_{11}a_{22}\,a_{33}a_{44}$$

$$A = \begin{bmatrix} a_{11} & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 \\ a_{31} & a_{32} & a_{33} & 0 \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \longrightarrow \det(A) = a_{11}a_{22}\,a_{33}a_{44}$$

- Jika A adalah matriks diagonal atau skalar, maka
  det(A) = a_{11} x a_{22} x … x a_{nn}
- Jika A adalah matriks persegi berordo *n* x *n* dan *k* adalah sembaran bilangan, maka
  det(*k*A) = k^n x det(A)
- Jika matriks A mempunyai invers atau *invertible*, maka
  det(A^{-1}) = $\frac{1}{\det(A)}$
- Jika A adalah matriks persegi yang memuat baris nol atau kolom nol, maka
  det(A) = 0
- Misalkan A adalah matriks persegi, kemudian A dikenakan operasi baris elementer, maka berlaku:

  3.1 Jika A^* diperoleh dari A dengan cara mengalikan satu baris dari A dengan sembarang bilangan $k \neq 0$ maka,

$\det(A^{*}) = k$ x det(A)

3.2 Jika A^{*} diperoleh dari A dengan cara menukar dua baris, maka

$\det(A^{*})$ = -det(A)

3.3 Jika A^{*} diperoleh dari A dengan cara menjumlahkan satu baris dengan kelipatan baris lain, maka

$\det(A^{*})$ = det(A)

Ada berbagai metode untuk mencari determinan diantaranya, yaitu metode sarrus, metode ekspansi kofaktor, metode reduksi baris sehingga terbentuk matriks segitiga atas atau bawah.

3.4 Determinan Matriks 2x2

det(A) = $a_{11}\, a_{22}$ - $a_{21}\, a_{22}$

3.5 Metode Sarrus

Untuk matriks berukuran 3x3 dapat menggunakan metode sarrus

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad \left[\begin{array}{ccc|cc} a_{11} & a_{12} & a_{13} & a_{11} & a_{12} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} \end{array}\right]$$

det(A) = ($a_{11}\, a_{22}\, a_{33} + a_{12}\, a_{23}\, a_{31} + a_{13}\, a_{21}\, a_{32}$) – ($a_{13}\, a_{22}\, a_{31} + a_{11}\, a_{23}\, a_{32} + a_{12}\, a_{21}\, a_{33}$)

3.6 Metode Reduksi Baris

Metode ini menggunakan operasi baris elementer hingga didapat matriks segitiga atas atau bawah.

$$[A] \overset{\text{OBE}}{\sim} [\text{matriks segitiga bawah}]$$

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

Maka det(A) = $(-1)^{p}$ x $a'_{11}\, a'_{22} \ldots a'_{nn}$ dengan p merupakan banyaknya operasi pertukaran baris yang dilakukan.

3.7 Metode Ekspansi Kofaktor

Metode ini menggunakan kofaktor dari salah satu baris atau kolom pada suatu matriks dikalikan dengan elemen matriks tersebut.

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

$$\begin{aligned} \det(A) &= a_{11}C_{11} + a_{12}C_{12} + \dots + a_{1n}C_{1n} \\ \det(A) &= a_{21}C_{21} + a_{22}C_{22} + \dots + a_{2n}C_{2n} \\ &\vdots \\ \det(A) &= a_{n1}C_{n1} + a_{n2}C_{n2} + \dots + a_{nn}C_{nn} \end{aligned}$$

Secara baris

$$\begin{aligned} \det(A) &= a_{11}C_{11} + a_{21}C_{21} + \dots + a_{n1}C_{n1} \\ \det(A) &= a_{12}C_{12} + a_{22}C_{22} + \dots + a_{n2}C_{n2} \\ &\vdots \\ \det(A) &= a_{1n}C_{1n} + a_{2n}C_{2n} + \dots + a_{nn}C_{nn} \end{aligned}$$

Secara kolom

2.8 Matriks Balikan

Jika A dan B merupakan sebuah matriks bujur sangkar sedemikian rupa sehingga AB = BA = I, maka B disebut matriks balikan dari A atau inverse dari A dan dapat dituliskan $B = A^{-1}$. Jika sebuah matriks determinannya sama dengan 0 maka matriks tersebut tidak memiliki matriks balikan atau inverse. Untuk mencari matriks balikan, kita bisa menggunakan rumus sebagai berikut.

$$A^{-1} = \frac{1}{\det(A)} \text{ x } \mathrm{Adj}(A)$$

2.9 Matriks Kofaktor

Matriks kofaktor adalah matriks yang tersusun dari nilai kofaktor setiap elemen matriks tersebut. Kofaktor dilambangkan dengan C_{ij} dan dapat dihitung dengan rumus:

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

Misalkan A adalah matriks *n* x *n* dan C_{ij} adalah kofaktor entri a_{ij}, maka matriks kofaktornya yaitu

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

M_{ij} di sini adalah minor dari dari setiap elemen matriks. M_{ij} ini merupakan determinan *submatrix* yang elemen-elemennya tidak berada pada baris i dan kolom j. Berikut merupakan contoh minor.

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

2.10 Matriks Adjoin

Matriks adjoin adalah transpose dari suatu matriks kofaktor atau suatu matriks yang elemennya merupakan kofaktor dari elemen-elemen matriks tersebut. Matriks adjoin ini dapat digunakan untuk mencari inverse dari sebuah matriks. Jika matriks C merupaka matriks kofaktor maka adjoin dari matriks C yaitu:

$$C = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \longrightarrow \text{adj}(C) = \begin{bmatrix} C_{11} & C_{21} & C_{31} \\ C_{12} & C_{22} & C_{32} \\ C_{13} & C_{23} & C_{33} \end{bmatrix}$$

2.11 Kaidah Cramer

Kaidah Cramer adalah rumus yang dapat digunakan untuk menyelesaikan sistem persamaan linear. Metode ini dinamai dari matematikawan Swiss Gabriel Cramer (1704–1752). Metode ini menggunakan determinan dari suatu matriks dan matriks lain yang diperoleh dengan mengganti salah satu kolom dengan vektor yang terdiri dari angka di ruas kanan persamaannya. Kaidah Cramer tidak efisien untuk sistem dengan lebih dari dua atau tiga persamaan. Kaidah Cramer juga tidak stabil secara numerik, termasuk untuk sistem 2×2.

Berikut ini merupakan rumus kaidah cramer.

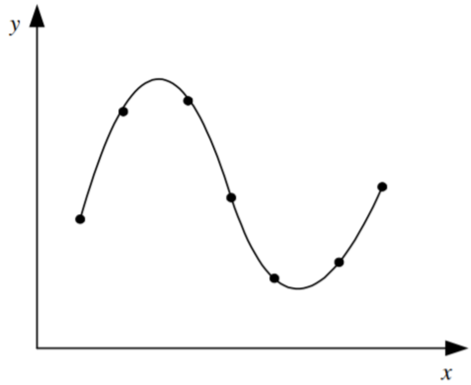
$$x_1 = \frac{\det(A_1)}{\det(A)}, \quad x_2 = \frac{\det(A_2)}{\det(A)}, \quad \ldots, \quad x_n = \frac{\det(A_n)}{\det(A)}$$

$$A = \begin{bmatrix} -1 & 2 & -3 \\ 2 & 0 & 1 \\ 3 & -4 & 4 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} \qquad \det(A) = \begin{vmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{vmatrix} = 10$$

$$A_1 = \begin{bmatrix} \mathbf{1} & 2 & -3 \\ \mathbf{0} & 0 & 1 \\ \mathbf{2} & -4 & 4 \end{bmatrix} \qquad A_2 = \begin{bmatrix} -1 & \mathbf{1} & -3 \\ 2 & \mathbf{0} & 1 \\ 3 & \mathbf{2} & 4 \end{bmatrix} \qquad A_3 = \begin{bmatrix} -1 & 2 & \mathbf{1} \\ 2 & 0 & \mathbf{0} \\ 3 & -4 & \mathbf{2} \end{bmatrix}$$

2.12 Interpolasi Polinom

Pencocokan kurva adalah sebuah metode yang mencocokkan titik data dengan sebuah kurva fungsi. Salah satu metode pencocokkan curva yaitu dengan interpolasi. Jika data diketahui mempunyai ketelitian yang sangat tinggi, maka kurva cocokannya dibuat melalui setiap titik. Kita katakana di sini bahwa kita menginterpolasi titik-titik data dengan sebuah fungsi. Bila fungsi cocokan yang digunakan berbentuk polinom, polinom tersebut dinamakan polinom interpolasi. Pekerjaan menginterpolasi titik data dengan sebuah polinom disebut interpolasi polinom. Contoh data yang berketelitian tinggi adalah titik-titik yang dihitung dari fungsi yang telah diketahui atau data tabel yang terdapat didalam acuan ilmiah, seperti data percepatan gravitasi bumi sebagai fungsi jarak sebuah titik ke pusat bumi. Selain dengan polinom, interpolasi titik-titik data dapat dilakukan dengan fungsi spline, fungsi rasional (pecahan), atau deret Fourier.

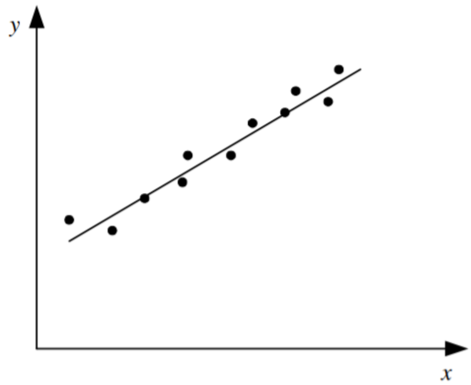


Aplikasi interpolasi polinom:

- Menghampiri fungsi rumit menjadi lebih sederhana
- Menggambar kurva (jika hanya diketahui titik-titik diskrit saja)

2.13 Regresi Linear Berganda

Selain interpolasi, regresi juga merupakan salah satu metode untuk pencocokan kurva. Data hasil pengukuran umumnya mengandung galat yang cukup berarti. Karena data ini tidak teliti, maka kurva yang mencocokan titik data itu tidak perlu melalui semua titik. Kurva tersebut cukup hanya mewakili kecenderungan titik data, yakni kurva mengikuti pola titik sebagai suatu kelompok.



Regresi

Regresi linear berganda merupakan model regresi linear yang melibatkan lebih dari satu variabel bebas. Regresi ini menggunakan interpolasi polinom untuk memprediksi nilai. Rumus dari regresi linear berganda yaitu:

$$Y = \alpha + \beta 1\ X2 + \beta 2\ X2 + \beta n\ Xn + e$$

Keterangan:

Y = Variabel terikat atau response.

X = Variabel bebas atau predictor.

α = Konstanta.

β = Slope atau Koefisien estimate.

BAB III
IMPLEMENTASI PROGRAM

3.1 Matriks.java

- Class: matriks

  Class ini berfungsi untuk menyelesaikan persamaan linear.

  1. Atribut
     a. baris: berfungsi untuk menyimpan jumlah baris dari suatu matriks.
     b. kolom: berfungsi untuk menyimpan jumlah kolom dari suatu matriks.
     c. tabFloat: berfungsi untuk menyimpan matriks.
  2. Method
     a. makeSPL: berfungsi untuk membuat objek matriks.
     b. setBaris: berfungsi untuk mengganti nilai baris dari matriks.
     c. setKolom: berfungsi untuk mengganti nilai kolom dari matriks.
     d. setElmt: berfungsi untuk mengganti nilai tabFloat di salah satu elemen matriks tersebut.
     e. setWholeTabFloat: berfungsi untuk mengganti semua tabFloat pada seluruh matriks tersebut.
     f. getBaris: berfungsi untuk mengambil nilai baris.
     g. getKolom: berfungsi untuk mengambil nilai kolom.
     h. getElmt: berfungsi untuk mengambil nilai pada suatu elemen matriks tersebut.
     i. getWholeTab: berfungsi untuk mengambil seluruh matriks.
     j. copyMatriks: berfungsi untuk menyalin matriks.
     k. divideAllbyLeading: berfungsi untuk membagi seluruh baris dengan element leading.
     l. exchangeRow: berfungsi untuk menukar baris pada matriks.
     m. searchNonZeroValue: berfungsi untuk mencari baris yang bisa ditukar.
     n. makeElmtBeforeLeadingToZero: berfungsi untuk mengurangi baris yang sedang dikerjakan dengan suatu baris yang lain agar element sebelum element leading menjadi nol.
     o. substractionRow: berfungsi untuk meng
     p. matriksToGauss: berfungsi untuk
     q. gaussToGaussJordan: berfungsi untuk

r. outputHasil: Mengeluarkan output final dari matriks yang sudah dilakukan OBE gauss/gauss-jordan

3.2 Inverse.java

- Class: Inverse

  Class ini berfungsi untuk menghasilkan matriks balikan atau inverse dari suatu matriks.

  1. Atribut
     a. baris: berfungsi untuk menyimpan jumlah baris dari suatu matriks.
     b. kolom: berfungsi untuk menyimpan jumlah kolom dari suatu matriks.
     c. tabFloat: berfungsi untuk menyimpan matriks.
  2. Method
     a. makeInverse: berfungsi untuk membuat objek inverse.
     b. setBaris: berfungsi untuk mengganti nilai baris dari matriks.
     c. setKolom: berfungsi untuk mengganti nilai kolom dari matriks.
     d. setElmt: berfungsi untuk mengganti nilai tabFloat di salah satu elemen matriks tersebut.
     e. setWholeTabFloat: berfungsi untuk mengganti semua tabFloat pada seluruh matriks tersebut.
     f. getBaris: berfungsi untuk mengambil nilai baris.
     g. getKolom: berfungsi untuk mengambil nilai kolom.
     h. getElmt: berfungsi untuk mengambil nilai pada suatu elemen matriks tersebut.
     i. getWholeTab: berfungsi untuk mengambil seluruh matriks.
     j. copyMatriks: berfungsi untuk menyalin matriks.
     k. Determinan: berfungsi untuk mendapatkan nilai determinan dari matriks.
     l. cofactor: berfungsi untuk membuat matriks kofaktor.
     m. transpose: berfungsi untuk men-transpose suatu matriks.
     n. cantInverse: berfungsi untuk mengecek apakah matriks tersebut mempunyai invers atau tidak.
     o. inverse: untuk men-inverse matriks.
     p. kaliMatriks: berfungsi untuk mengalikan dua buah matriks.

3.3 cramer.java

- Class: cramer

  Class ini berfungsi untuk menyelesaikan persamaan linear menggunakan kaidah cramer.

  1. Atribut

a. baris: berfungsi untuk menyimpan jumlah baris dari suatu matriks.
b. kolom: berfungsi untuk menyimpan jumlah kolom dari suatu matriks.
c. tabFloat: berfungsi untuk menyimpan matriks.

2. Method
   a. makeCramer: berfungsi untuk membuat objek cramer.
   b. setBaris: berfungsi untuk mengganti nilai baris dari matriks.
   c. setKolom: berfungsi untuk mengganti nilai kolom dari matriks.
   d. setElmt: berfungsi untuk mengganti nilai tabFloat di salah satu elemen matriks tersebut.
   e. setWholeTabFloat: berfungsi untuk mengganti semua tabFloat pada seluruh matriks tersebut.
   f. getBaris: berfungsi untuk mengambil nilai baris.
   g. getKolom: berfungsi untuk mengambil nilai kolom.
   h. getElmt: berfungsi untuk mengambil nilai pada suatu elemen matriks tersebut.
   i. copyMatriks: berfungsi untuk menyalin matriks.
   j. switchColumn: menukar kolom
   k. squareMatrix: membuat dari matriks mentah menjadi matriks persegi
   l. determinan: berfungsi untuk mendapatkan nilai determinan suatu matriks.
   m. cramerMethod: proses cramer berlangsung
   n. printHasil: hasil cramer dikeluarkan
   o. printTxt:

3.4 interpolasi.java

- Class: interpolasi

  Class ini berfungsi untuk

  1. Atribut
     a. baris: berfungsi untuk menyimpan jumlah baris dari suatu matriks.
     b. kolom: berfungsi untuk menyimpan jumlah kolom dari suatu matriks.
     c. tabFloat: berfungsi untuk menyimpan matriks.
  2. Method
     a. makeInterpolasi: berfungsi untuk membuat objek interpolasi.
     b. setBaris: berfungsi untuk mengganti nilai baris dari matriks.
     c. setKolom: berfungsi untuk mengganti nilai kolom dari matriks.
     d. setElmt: berfungsi untuk mengganti nilai tabFloat di salah satu elemen matriks

tersebut.

e. setWholeTabFloat: berfungsi untuk mengganti semua tabFloat pada seluruh matriks tersebut.
f. getBaris: berfungsi untuk mengambil nilai baris.
g. getKolom: berfungsi untuk mengambil nilai kolom.
h. getElmt: berfungsi untuk mengambil nilai pada suatu elemen matriks tersebut.
i. copyInterpolasi: mengcopy interpolasi 1 ke yang lain
j. convertToMatriks: merubah dari matriks titik menjadi matriks persamaan linier
k. outputInterpolasi: mengeluarkan hasil final interpolasi

3.5 regresi.java

- Class: regresi

  Class ini berfungsi untuk menghitung regresi linier berganda dari inputan

  1. Atribut
     a. baris: berfungsi untuk menyimpan jumlah baris dari suatu matriks.
     b. kolom: berfungsi untuk menyimpan jumlah kolom dari suatu matriks.
     c. tabX: untuk matriks persamaan.
     d. tabY: untuk matriks hasil dari persamaan.
     e. nilaix: untuk nilai x yang akan dipakai untuk menaksir persamaan.
     f. tabFloat: berfungsi untuk menyimpan matriks.
  2. Method
     a. makeRegresi: berfungsi untuk membuat inputannya.
     b. kaliMatriks: berfungsi untuk mengalikan dua buah matriks.
     c. transpose: berfungsi untuk men-transpose matriks.
     d. makeSigmaX: berfungsi untuk membuat sigma X
     e. makeSigmaY: berfungsi untuk membuat sigma Y.
     f. makeSPL: untuk menggabungkan sigma X dan sigma Y jadi sistem persamaan linear.
     g. determineHasil: untuk mencetak hasil ke layer.

3.6 Main.java

- Class: Main

  Class ini berfungsi untuk menjadi gerbang pusat dari kumpulan class pada package SPL

  1. Atribut: Tidak ada
  2. Method

a. main: menjadi gerbang utama program

3.7 Menu.java

- Class: Menu

  Class ini berfungsi untuk mempercantik program pada main sehingga program main tidak terlihat berantakan

  1. Atribut: Tidak ada
  2. Method:
     a. SPL: mencangkup semua menu dari SPL
     b. Determinan: mencangkup semua menu dari Determinan
     c. MatriksBalikan: mencanhgkup semua menu dari Matriks Balikan
     d. Interpolasi: mencangkup semua menu dari Interpolasi
     e. Regresi: mencnagkup semua menu dari regresi

3.8 Input.java

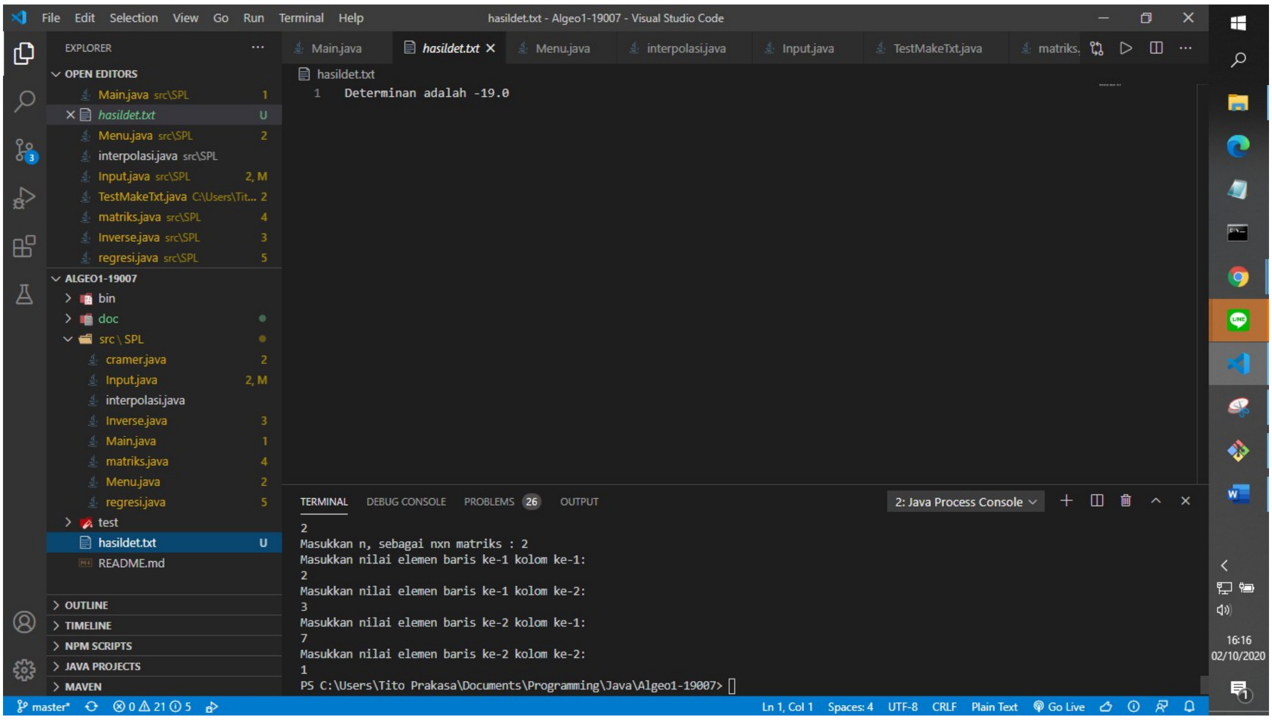
- Class: Input

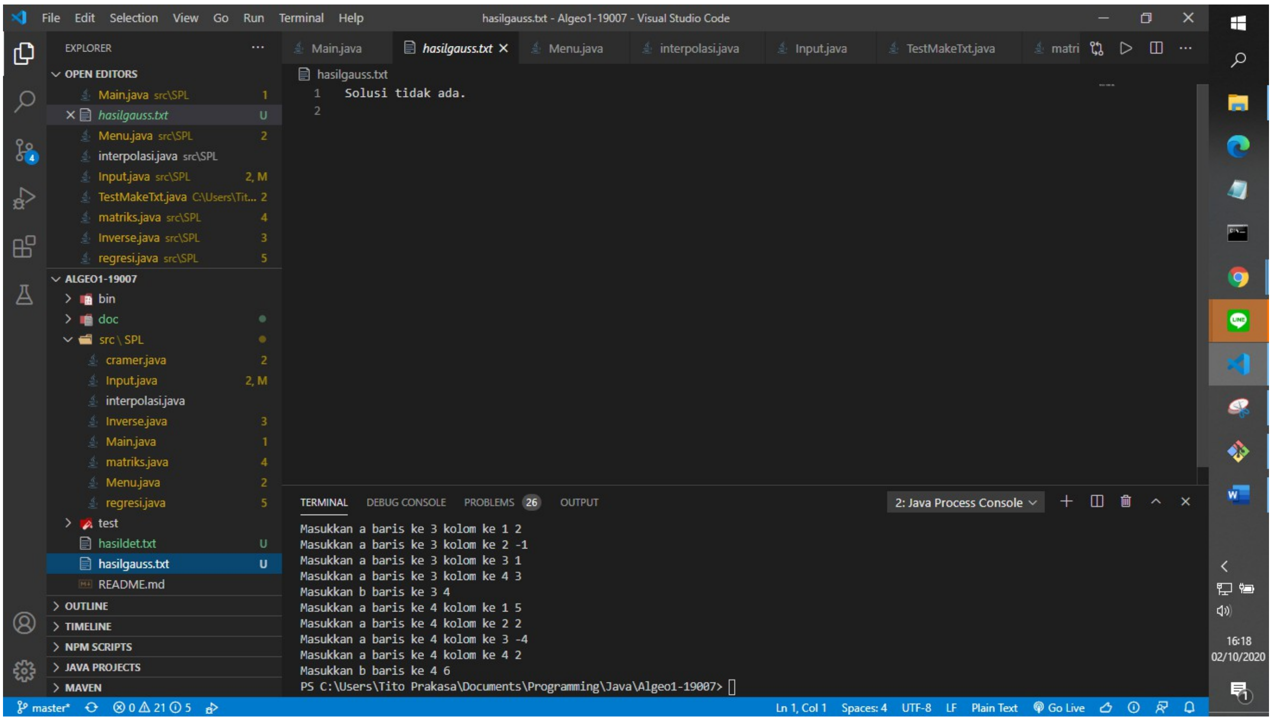
  Class ini berfungsi untuk mempermudah masukan dari txt file ke program

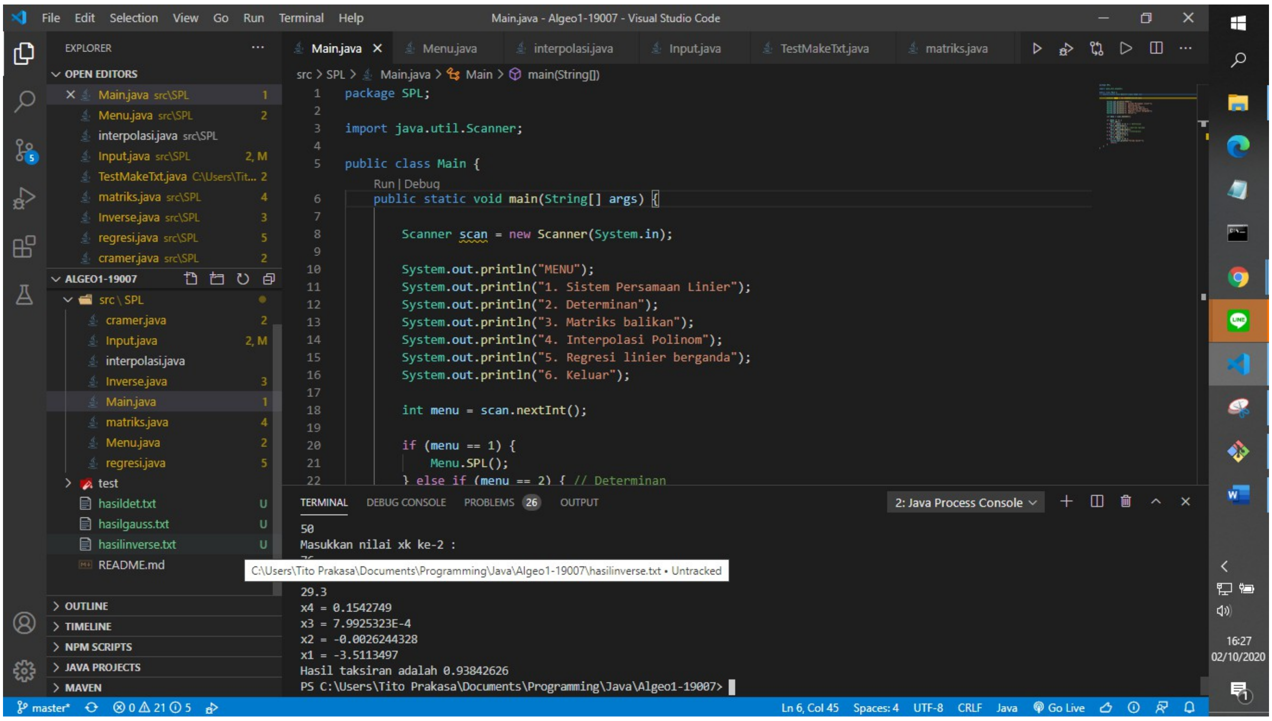
  3. Atribut: Tidak ada
  4. Method:
     a. menghitungKolom: method antara untuk mempermudah membuat matriks dari file txt ke program
     b. menghitungBaris: method antara juga untuk mempermudah membuat matriks dari file txt ke program
     c. input: method utama sebagai perantara antara file ke program.

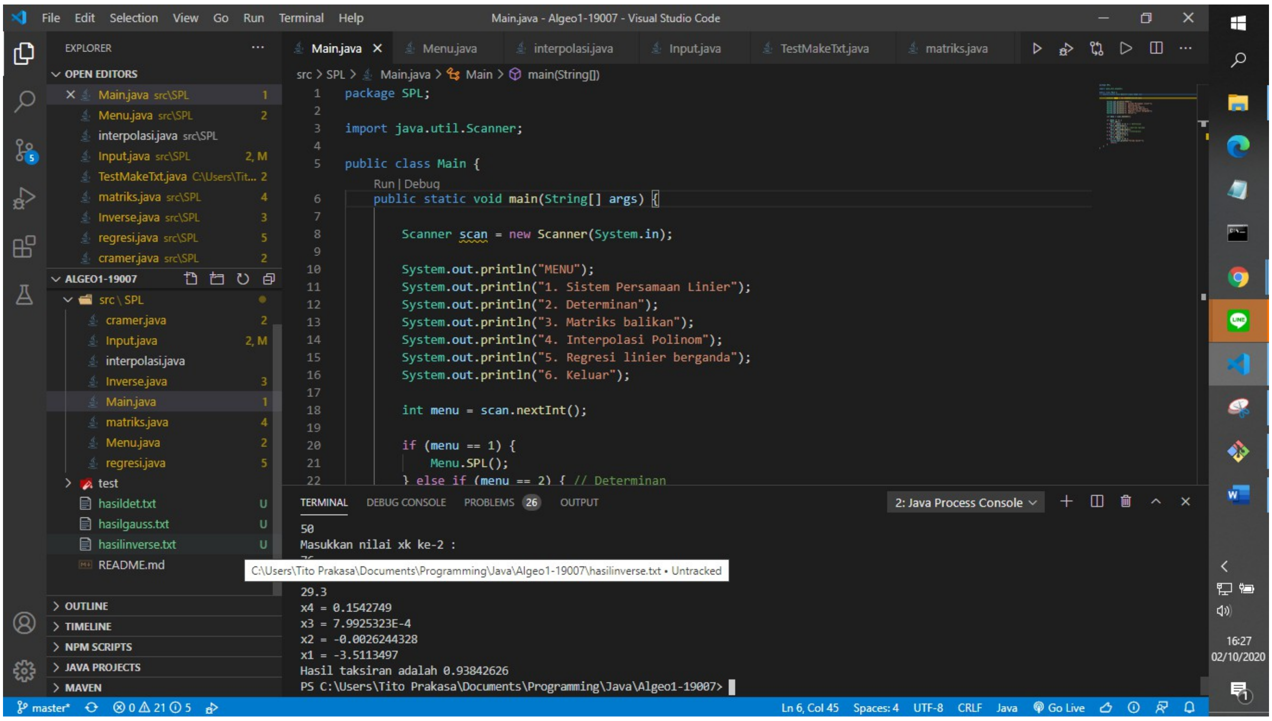
BAB IV

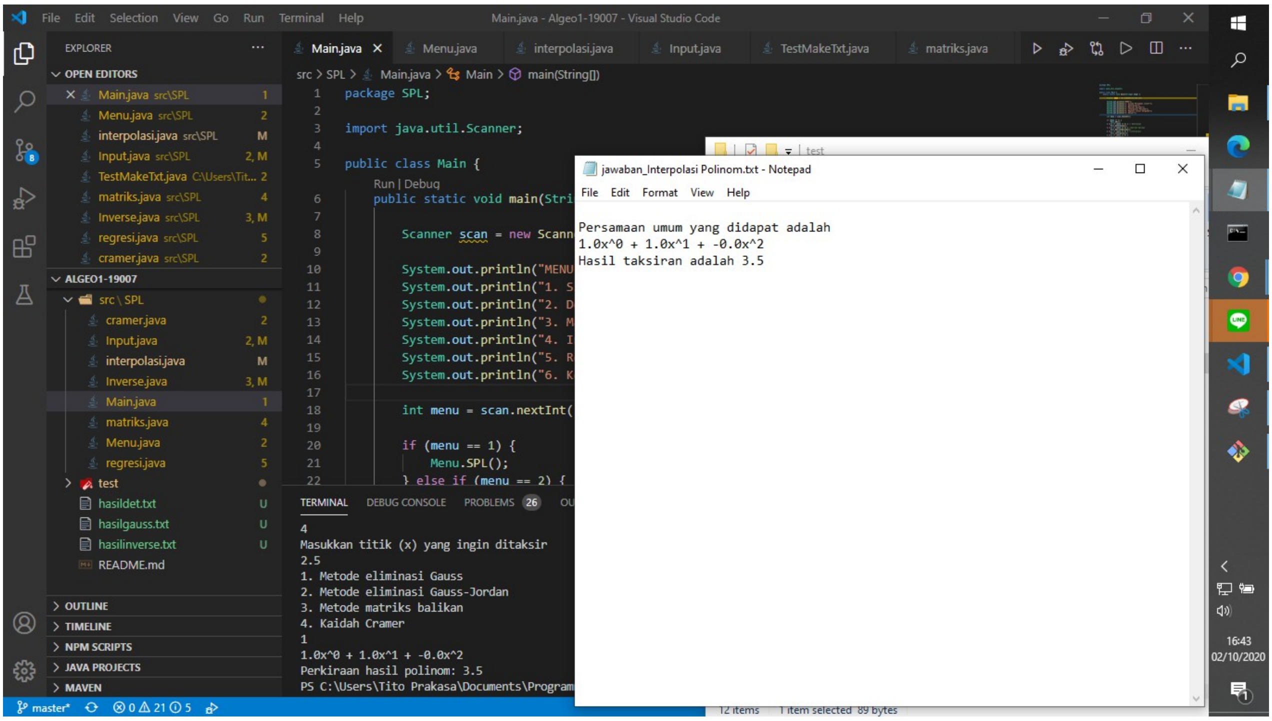
IMPLEMENTASI PROGRAM






Main.java - Algeo1-19007 - Visual Studio Code
package SPL;
import java.util.Scanner;
public class Main {
public static void main(String[] args) {
Scanner scan = new Scanner(System.in);
System.out.println("MENU");
System.out.println("1. Sistem Persamaan Linier");
System.out.println("2. Determinan");
System.out.println("3. Matriks balikan");
System.out.println("4. Interpolasi Polinom");
System.out.println("5. Regresi linier berganda");
System.out.println("6. Keluar");
int menu = scan.nextInt();
if (menu == 1) {
Menu.SPL();
} else if (menu == 2) { // Determinan
50
Masukkan nilai xk ke-2 :
C:\Users\Tito Prakasa\Documents\Programming\Java\Algeo1-19007\hasilinverse.txt • Untracked
29.3
x4 = 0.1542749
x3 = 7.9925323E-4
x2 = -0.0026244328
x1 = -3.5113497
Hasil taksiran adalah 0.93842626
PS C:\Users\Tito Prakasa\Documents\Programming\Java\Algeo1-19007>


Main.java - Algeo1-19007 - Visual Studio Code
package SPL;
import java.util.Scanner;
public class Main {
public static void main(String[] args) {
Scanner scan = new Scanner(System.in);
System.out.println("MENU");
System.out.println("1. Sistem Persamaan Linier");
System.out.println("2. Determinan");
System.out.println("3. Matriks balikan");
System.out.println("4. Interpolasi Polinom");
System.out.println("5. Regresi linier berganda");
System.out.println("6. Keluar");
int menu = scan.nextInt();
if (menu == 1) {
Menu.SPL();
} else if (menu == 2) { // Determinan
50
Masukkan nilai xk ke-2 :
C:\Users\Tito Prakasa\Documents\Programming\Java\Algeo1-19007\hasilinverse.txt • Untracked
29.3
x4 = 0.1542749
x3 = 7.9925323E-4
x2 = -0.0026244328
x1 = -3.5113497
Hasil taksiran adalah 0.93842626
PS C:\Users\Tito Prakasa\Documents\Programming\Java\Algeo1-19007>


Persamaan umum yang didapat adalah
1.0x^0 + 1.0x^1 + -0.0x^2
Hasil taksiran adalah 3.5
Masukkan titik (x) yang ingin ditaksir
2.5
1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer
1.0x^0 + 1.0x^1 + -0.0x^2
Perkiraan hasil polinom: 3.5

BAB V

KESIMPULAN, SARAN, DAN REFLEKSI

5.1 **Kesimpulan**

Program yang telah kami buat berhasil jadi, tetapi masih kurang rapi atau sedikit berantakan. Selain itu juga, dalam program kami masih ada fitur yang belum terealisasikan. Penerapan algoritma penyelesaian SPL dapat diterapkan di Java.

5.2 **Saran**

Sebaiknya pengelolaan file lebih dirapikan lagi dan algoritmanya lebih diringkas lagi agar lebih sederhana.

5.3 **Refleksi**

Seharusnya kami dapat lebih baik lagi dalam melakukan pembagian tugasnya dan lebih terstruktur serta harus dari jauh hari dalam mengerjakannya sehingga tugas besar ini dapat dibuat dengan matang.

DAFTAR PUSTAKA

- Howard Anton, Elementary Linear Algebra, 10th edition, John Wiley amnd Sons, 2010
- https://www.profematika.com/10-sifat-determinan-dan-reduksi-baris-beserta-contohnya/
- https://id.wikipedia.org/wiki/Determinan#:~:text=Dalam%20bidang%20aljabar%20linear%2C%20determinan,transformasi%20yang%20digambarkan%20oleh%20matriks.
- https://id.wikipedia.org/wiki/Sistem_persamaan_linear
- https://id.wikipedia.org/wiki/Eliminasi_Gauss
- https://id.wikipedia.org/wiki/Aljabar_linear#:~:text=Matriks%20Balikan%20(Invers),-Orde%202x2&text=Jika%20tidak%20ditemukan%20matriks%20B,dari%20A%20maka%20B%20%3D%20C.&text=Karena%20AB%20%E2%89%A0%20BA%20%E2%89%A0,matriks%20B%20disebut%20matriks%20tunggal.