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

Oleh:
Rifqi Naufal Abdjul (13520062)
Amar Fadil (13520103)
Vito Ghifari (13520153)

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2021

DAFTAR ISI

BAB 1
Deskripsi Masalah

I. Tujuan

Membuat program dalam bahasa Java untuk:

1. 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)
2. Menghitung determinan matriks dengan reduksi baris dan dengan ekspansi kofaktor
3. Menghitung balikan matriks
4. Menyelesaikan persoalan interpolasi dan regresi linier,

II. Spesifikasi

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 10
   -3 7 8.3 11
   0.5 -10 -9 12

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), 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. Untup 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
7. Untuk persoalan polinom interpolasi dan regresi, luarannya adalah persamaan polinom/regresi dan taksiran nilai fungsi pada x yang diberikan.
8. Luaran program harus dapat ditampilkan pada layar komputer dan dapat disimpan ke dalam file.
9. Bahasa program yang digunakan adalah Java.
10. Program tidak harus berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas *Eclipse* misalnya).
11. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

    MENU
    1. Sistem Persamaaan Linier
    2. Determinan
    3. Matriks balikan
    4. Interpolasi Polinom
    5. Regresi linier berganda
    6. 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

Matriks

a. Definisi

Matriks adalah kumpulan bilangan, simbol, atau ekspresi yang disusun secara baris dan kolom di dalam suatu tanda kurung. Sebuah matriks mempunyai ukuran yaitu, jumlah baris x jumlah kolomnya. Con: Matriks 3x2, berarti matriks dengan 3 baris dan 2 kolom.

b. Jenis Matriks

Terdapat beberapa jenis matriks berdasarkan ukuran dan bilangan yang menyusunnya.

- Matriks Persegi
  Merupakan matriks yang mempunyai jumlah baris dan kolom yang sama.
- Matriks Identitas
  Merupakan matriks yang bernilai 0, selain pada diagonalnya yang bernilai 1.
- Matriks segitiga
  - Matriks segitiga atas
    Merupakan matriks yang bernilai 0 pada bagian segitiga bawah (di bawah diagonal) dan bernilai bebas pada bagian lainnya.
  - Matriks segitiga bawah
    Merupakan matriks yang bernilai 0 pada bagian segitiga atas (di atas diagonal) dan bernilai bebas pada bagian lainnya.
- Matriks eselon
  Matriks yang berkait hanyalah matriks eselon baris yang dibagi menjadi 2, yaitu:
  - Matriks eselon baris
    Merupakan matriks yang mempunyai bilangan 1 utama pada tiap baris dan berurut dari atas (1 utama paling depan) ke bawah (1 utama paling belakang) dan angka yang boleh menempati sebelum angka 1 utama adalah 0.
  - Matriks eselon baris tereduksi
    Merupakan matriks eselon baris yang pada baris yang mengandung 1 utama, hanya memiliki angka 0 dan 1 utama itu tersebut.

c. Operasi Dasar Matriks

Terdapat beberapa operasi dasar matriks, yaitu:

- Penjumlahan dan pengurangan
  Operasi penjumlahan dan pengurangan pada matriks mempunyai prekondisi bahwa kedua matriks harus berbentuk sama (jumlah baris dan jumlah kolom sama). Operasi penjumlahan atau pengurangan dilakukan dengan cara menambahkan atau mengurangkan tiap elemen sesuai dengan posisi baris dan kolomnya.
- Perkalian skalar
  Operasi perkalian skalar pada matriks dilakukan dengan mengalikan seluruh elemen matriks dengan konstanta skalar.

- Perkalian antar matriks
  Operasi perkalian antar matriks mempunyai prekondisi bahwa jumlah kolom matriks pertama harus sama dengan jumlah baris matriks kedua dan akan menghasilkan matriks dengan jumlah baris matriks pertama dan jumlah kolom matriks kedua. Perkalian antar matriks dilakukan dengan cara mengalikan baris ke-n matriks pertama dan kolom ke-m matriks kedua dan hasilnya akan menjadi elemen ke n,m pada matriks hasil.
- Transpose matriks
  Transpose matriks merupakan operasi pada matriks yang menukar semua elemen baris menjadi elemen kolom dan juga sebaliknya. Sebuah matriks yang berukuran m x n akan menjadi n x m jika dilakukan transpose pada matriks tersebut.

Operasi Baris Elementer

Operasi baris elementer (OBE) merupakan operasi yang mempengaruhi baris pada suatu matriks. Operasi ini biasanya tidak mengubah identitas seperti determinan dan biasa dilakukan untuk menyederhanakan matriks menjadi matriks eselon baris. Terdapat 3 jenis OBE, yaitu:

a. Penjumlahan baris

Menjumlahkan baris ke baris yang lain (bukan baris sendiri) dengan perkalian skalar baris yang ditambahkan. Operasi ini tidak mengubah determinan sama sekali.
Con:

$$\begin{bmatrix} 1 & 3 & 2 \\ 6 & 5 & 4 \\ 7 & 7 & 7 \end{bmatrix} \xrightarrow[R1 + 2 \cdot R2]{} \begin{bmatrix} 13 & 13 & 10 \\ 6 & 5 & 4 \\ 7 & 7 & 7 \end{bmatrix}$$

b. Pertukaran baris

Menukar baris dengan baris yang lain (bukan baris sendiri). Operasi ini mengubah determinan dengan perubahan $D^* = D\,.\,(-\,1)$
Con:

$$\begin{bmatrix} 1 & 3 & 2 \\ 6 & 5 & 4 \\ 7 & 7 & 7 \end{bmatrix} \xrightarrow[R1\; swap\; R2]{} \begin{bmatrix} 6 & 5 & 4 \\ 1 & 3 & 2 \\ 7 & 7 & 7 \end{bmatrix}$$

c. Perkalian skalar baris

Mengalikan suatu baris dengan suatu bilangan skalar. Operasi ini mengubah determinan dengan perubahan $D^* = D\,.\,\frac{1}{k}$
Con:

$$\begin{bmatrix} 1 & 3 & 2 \\ 6 & 5 & 4 \\ 7 & 7 & 7 \end{bmatrix} \xrightarrow[R3\, \cdot\, 1/7]{} \begin{bmatrix} 1 & 3 & 2 \\ 6 & 5 & 4 \\ 1 & 1 & 1 \end{bmatrix}$$

Eliminasi Gauss

Eliminasi gauss merupakan eliminasi menggunakan operasi baris elementer untuk menghasilkan matriks eselon baris atau eselon baris tereduksi. Eliminasi gauss ini biasa digunakan untuk menyelesaikan Sistem Persamaan Linear (SPL) banyak peubah yang dibentuk menjadi *augmented matrix*. Eliminasi Gauss dibedakan menjadi 2 metode, yaitu:

- Gauss Method
  Merupakan metode eliminasi gauss yang menghasilkan matriks eselon baris.
- Gauss-Jordan Method
  Merupakan metode eliminasi gauss yang menghasilkan matriks eselon baris tereduksi.

Setelah suatu matriks diubah menjadi matriks eselon baris, matriks tersebut diperiksa kekonsistenannya. Jika matriks yang terbentuk tidak konsisten, SPL tidak mempunyai solusi. Jika matriks tersebut konsisten, *backward substitution* dilakukan agar solusi dari setiap variabel didapatkan.

Ekspansi Kofaktor

Ekspansi kofaktor merupakan metode untuk menghitung determinan matriks berukuran n x n dengan matriks minornya, yang merupakan determinan dari submatriks tersebut. Ekspansi kofaktor dirumuskan sebagai berikut,

$$\det(B) = \sum_{j=1}^{n} (-1)^{i+j} B_{i,j} M_{i,j}$$

dimana,
$B_{i,j}$ = Elemen B pada baris ke i dan kolom ke j
$M_{i,j}$ = Submatriks B yang didapat dari menghapus baris ke-i dan kolom ke-j

Kaidah Cramer

Merupakan metode untuk menyelesaikan sistem persamaan linear dengan banyak persamaan dan variabel. Metode ini mengubah persamaan linear menjadi sebuah matriks yang memenuhi persamaan Ax = B. Kaidah cramer menjelaskan bahwa untuk setiap solusi x, berlaku:

$$x_i = \frac{|\Delta_i|}{|\Delta|}$$

dimana
Δ = Matriks utama
Δ_i = Matriks yang dibentuk oleh hasil substitusi kolom ke-i dari matriks A dengan B

Matriks Balikan

Merupakan invers dari suatu matriks yang memenuhi $A.A^{-1} = 1$. Terdapat beberapa metode yang bisa digunakan untuk membentuk matriks balikan, seperti:

1. Metode Gauss-Jordan
   Melakukan operasi baris elementer hingga membentuk matriks identitas yang berukuran sama. Lalu melakukan operasi baris elementer dengan urutan yang sama terhadap matriks identitas untuk menghasilkan matriks balikan.

2. Metode Adjoint
   Mencari matriks kofaktor dari matriks yang ingin di-inverskan. Matriks kofaktor adalah matriks yang mempunyai nilai sama dengan determinan dari matriks minor yang dibentuk setiap elemen matriks utama. Lalu, matriks kofaktor akan di transpose membentuk matriks adjoint yang akan dibagi dengan determinan dari matriks utama untuk menghasilkan matriks balikan.

Matriks tidak akan memiliki invers jika determinan dari matriks tersebut bernilai 0 karena, pada metode gauss-jordan tidak dapat menghasilkan matriks identitas dan pada metode adjoint akan terjadi pembagian dengan nilai 0.

Interpolasi Polinom

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 $i = 0, 1, 2, …, n$.

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 $p_n(x) = a_0 + a_1x + a_2x^2 + … + a_nx^n$. 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 $p_n(x) = a_0 + a_1x + a_2x^2 + … + a_nx^n$ untuk $i = 0, 1, 2, …, n$, akan diperoleh n buah sistem persamaan lanjar dalam $a_0 , a_1 , a_2 , …, a_n$,

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

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 sistem 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 2 titik tersebut adalah p2(x) = 0.6762 + 0.2266x - 0.0064x 2 . Dengan menggunakan polinom ini, maka nilai fungsi pada x = 9.2 dapat ditaksir sebagai berikut: p2(9.2) = 0.6762 + 0.2266(9.2) - 0.0064(9.2)2 = 2.2192.

Regresi Linear Berganda

Regresi linear adalah sebuah metode pendekatan yang melibatkan satu atau lebih variabel independen untuk mendapatkan hubungan antara variabel tersebut dengan satu jenis variabel yang dihasilkan (variabel dependen). Regresi linear terbagi menjadi dua, yaitu regresi linear sederhana dan regresi linear berganda. Regresi linear sederhana hanya melibatkan satu jenis variabel independen, sedangkan regresi linear berganda terdiri atas dua variabel independen atau lebih. Hasil regresi linear berganda dapat dinyatakan dalam

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

dengan

- y_i adalah variabel dependen untuk *observation* ke-i
- β_0 adalah *intercept*, yaitu nilai y saat seluruh x sama dengan nol
- β_k adalah koefisien untuk x yang bersangkutan, yaitu x_k
- x_{ki} adalah variabel independen ke-k untuk *observation* ke-i
- ϵ_i adalah *error-term* untuk *observation* ke-i.

Namun, pemecahan masalah ini dicari solusinya dengan prediksi. Artinya, diasumsikan bahwa *error-term* bernilai nol untuk setiap *observation*. Dengan demikian, persamaan akhirnya menjadi

$$\hat{y}_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_k x_{ki}$$

dengan $\hat{y}_i$ adalah hasil prediksi dari variabel dependen.

Adapun persamaan akhir di atas didapatkan dari persamaan

$$A\hat{x} = b$$

dengan A, x̂, dan b masing-masing adalah sebagai berikut.

$$A = \begin{bmatrix} 1 & x_{11} & \dots & x_{k1} \\ 1 & x_{12} & \dots & x_{k2} \\ \dots & \dots & \dots & \dots \\ 1 & x_{1n} & \dots & x_{kn} \end{bmatrix} \hat{x} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \dots \\ \beta_n \end{bmatrix} b = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix}$$

Untuk mendapatkan solusi dari sistem persamaan linear sebanyak *n* buah peubah, dilakukan operasi sebagai berikut.

$$A\hat{x} = b$$

$$A^T A\hat{x} = A^T b$$

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{11} & x_{12} & \dots & x_{1n} \\ \dots & \dots & \dots & \dots \\ x_{k1} & x_{k2} & \dots & x_{kn} \end{bmatrix} \begin{bmatrix} 1 & x_{11} & \dots & x_{k1} \\ 1 & x_{12} & \dots & x_{k2} \\ \dots & \dots & \dots & \dots \\ 1 & x_{1n} & \dots & x_{kn} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \dots \\ \beta_n \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{11} & x_{12} & \dots & x_{1n} \\ \dots & \dots & \dots & \dots \\ x_{k1} & x_{k2} & \dots & x_{kn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix}$$

Pada akhirnya, akan terbentuk matriks *augmented* yang dapat dicari solusi untuk *β* dengan metode Gauss ataupun Gauss-Jordan.

$$\begin{bmatrix} n\beta_0 & \beta_1 \sum_{i=1}^{n} x_{1i} & \beta_2 \sum_{i=1}^{n} x_{2i} & \dots & \beta_k \sum_{i=1}^{n} x_{ki} & \sum_{i=1}^{n} y_i \\ \beta_0 \sum_{i=1}^{n} x_{1i} & \beta_1 \sum_{i=1}^{n} x_{1i}^2 & \beta_2 \sum_{i=1}^{n} x_{1i} x_{2i} & \dots & \beta_k \sum_{i=1}^{n} x_{1i} x_{ki} & \sum_{i=1}^{n} x_{1i} y_i \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \beta_0 \sum_{i=1}^{n} x_{ki} & \beta_1 \sum_{i=1}^{n} x_{ki} x_{1i} & \beta_2 \sum_{i=1}^{n} x_{ki} x_{2i} & \dots & \beta_k \sum_{i=1}^{n} x_{ki}^2 & \beta_1 \sum_{i=1}^{n} x_{ki} y_i \end{bmatrix}$$

BAB 3
Implementasi

Kami membuat program kami berdasarkan beberapa packages yang memuat beberapa modul yang mempunyai class dan method sesuai dengan fungsinya masing masing. Berikut adalah penjelesan dari implementasi program kami.

1. Package Matriks

1.1. Modul Matriks (Matrix.java)

1.1.1. Class Matrix

Merupakan ADT dasar yang digunakan dalam keseluruhan program.

1.1.1.1. Methods

1.1.1.1.1. Konstruktor

Membentuk matriks kosong dengan input jumlah kolom dan jumlah baris atau membentuk matriks dengan input array 2 dimensi.

1.1.1.1.2. Selektor

Methods untuk mendapatkan dan menetapkan nilai dari atribut matriks.

1.1.1.1.3. isAlmostEqElmt

Untuk membandingkan nilai elemen dengan perbedaan sekitar 10^-5.

1.1.1.1.4. isSquare

Untuk memeriksa apakah matriks berbentuk persegi atau tidak.

1.1.1.1.5. isIdentity

Untuk memeriksa apakah matriks merupakan matriks identitas atau tidak.

1.1.1.1.6. isEchelon

Untuk memeriksa apakah matriks merupakan matriks eselon atau tidak.

1.1.1.1.7. isEchelonReduced

Untuk memeriksa apakah matriks merupakan matriks eselon tereduksi atau tidak.

1.1.1.1.8. isTriangular

Untuk memeriksa apakah matriks termasuk matriks segitiga atas, segitiga bawah, diagonal, atau tidak sama sekali.

1.1.1.1.9. add

Untuk melakukan operasi pertambahan terhadap 2 matriks.

1.1.1.1.10. sub

Untuk melakukan operasi pengurangan terhadap 2 matriks.

1.1.1.1.11. mul

Untuk melakukan operasi perkalian terhadap 2 matriks.

1.1.1.1.12. transpose

Untuk men-transpose matriks (menukar baris menjadi kolom, kolom menjadi baris).

1.1.1.1.13. RowSum

Untuk melakukan operasi baris elementer yaitu menambahkan baris ke baris lainnya dalam kelipatan konstanta.

1.1.1.1.14. RowSwap

Untuk melakukan operasi baris elementer yaitu menukar posisi kedua baris.

1.1.1.1.15. ScalarRowMultiplication

Untuk melakukan operasi baris elementer yaitu mengalikan baris dengan nilai skalar.

1.1.1.2. Atribut

1.1.1.2.1. NCol (integer)

Berisikan jumlah kolom matriks.

1.1.1.2.2. NRow (integer)

Berisikan jumlah baris matriks.

1.1.1.2.3. data (double[][])

Berisikan array 2 dimensi yang berisikan data matriks berukuran NCol dan NRow.

1.2. Modul Util (Util.java)

1.2.1. Class Util

Merupakan class pembantu (helper) dalam melakukan formatting, rounding, dan comparing floating point.

1.2.1.1. Methods

1.2.1.1.1. format

Untuk melakukan formatting string berdasarkan mode yang dipilih oleh user. Terdapat 4 mode formatting yaitu SHORT (3 angka dibelakang koma, opsional), LONG (13 angka dibelakang koma, opsional), PADDED_SHORT (3 angka dibelakang koma, padded dengan 0), dan DEFAULT (String.valueOf())..

1.2.1.1.2. roundDec

Untuk melakukan pembulatan angka sebesar jumlah angka di belakang koma.

1.2.1.1.3. isAlmostEq

Untuk membandingkan 2 angka dengan perbandingan relatif..

2. Package Solver

2.1. Modul SPLSolver (SPLSolver.java)

2.1.1. Class SPLSolver

Merupakan class yang dapat menyelesaikan permasalahan Sistem Persamaan Linear.

2.1.1.1. Methods

2.1.1.1.1. Gauss

Melakukan perhitungan eliminasi Gauss terhadap matriks augmented dengan OBE sehingga membentuk eselon kemudian melakukan *backward substitution* untuk mendapatkan solusi (jika ada) dari SPL.

2.1.1.1.2. Gauss-Jordan

Melakukan perhitungan eliminasi Gauss-Jordan terhadap matriks augmented dengan OBE sehingga membentuk eselon tereduksi kemudian melakukan *backward substitution* untuk mendapatkan solusi (jika ada) dari SPL.

2.1.1.1.3. backwardSubstitution

Melakukan substitusi dengan *dependency matrix*, yaitu membuat sebuah matriks A x A+1 dengan A jumlah peubah pada SPL tersebut yang merepresentasikan nilai pada matriks eselon. Kemudian, untuk setiap baris ke-i akan dilakukan toggle state peubah menjadi solusi, kemudian substitusi tiap koefisien dependan jika koefisien peubah bukan parametrik dan bukan nol, sehingga didapatkan *dependency matrix* dengan koefisien peubah bukan parametrik adalah nol. Terakhir, melakukan kalkulasi penghitung peubah yang akan menjadi parametrik (jika ada), lalu mengembalikan SolutionResult dengan *state* dan *dependency matrix* yang telah dibuat. Dilakukan juga pengecekan baris yang semua elemennya 0 kecuali pada konstanta (kolom terakhir) sehingga mengembalikan SolutionResult dengan SolutionType.INCONSISTENT.

2.1.1.1.4. Inverse

Melakukan perhitungan pada matriks augmented Ax=b dengan A merupakan matriks persegi, sehingga mendapatkan solusi dari x dengan membalikkan (inverse) matriks A kemudian dikali dengan b. Jika tidak terdapat balikan A atau A bukan persegi, akan mengembalikan SolutionResult dengan tipe SolutionType.NOT_SUPPORTED.

2.1.1.1.5. Crammer

Melakukan perhitungan pada matriks augmented Ax=b dengan A merupakan matriks persegi, sehingga mendapatkan solusi dari xi dengan mencari determinan matriks hasil substitusi b pada kolom A ke-i dibagi dengan determinan A. Jika determinan A adalah 0, Jika tidak terdapat balikan A atau A bukan persegi, akan mengembalikan SolutionResult dengan tipe SolutionType.NOT_SUPPORTED.

2.1.2. Class SolutionResult

Merupakan immutable class yang menampung hasil solusi dari SPL setelah dikalkulasi oleh metode pada SPLSolver.

2.1.2.1. Methods

2.1.2.1.1. generateVarString

Menghasilkan nama variabel parametrik berdasarkan argumen num dengan format `$huruf$angka` hasil kalkulasi chr(‘a’+(num%26)) + str(num/27). Jika num/27 = 0, bagian angka tidak akan ditulis.

2.1.2.2. Atribut

2.1.2.2.1. result (getResult)

Hasil dependency matrix. Akan dikompresi jika solusi unik menjadi matriks N x 1 dengan N merupakan banyak peubah dari SPL.

2.1.2.2.2. states (getStates)

State dari setiap variabel. Jika

2.1.2.2.3. type (getType)

Tipe dari solusi. Merupakan tipe enumerasi SolutionType dengan 4 nilai: NOT_SUPPORTED, INCONSISTENT, INFINITE, dan UNIQUE.

2.1.2.2.4. intermediate (getIntermediate)

Menyimpan matriks intermediasi yang didapatkan sebelum diproses menjadi hasil solusi. Akan null jika menggunakan metode Crammer atau tipe solusi NOT_SUPPORTED dan INCONSISTENT..

2.2. Modul DeterminantSolver (DeterminantSolver.java)

2.2.1. Class DeterminantSolver

Merupakan class penghitung determinan berdasarkan input matriks.

2.2.1.1. Methods

2.2.1.1.1. CofactorExpansion

Melakukan perhitungan determinan secara rekursif dengan mereduksi matriks menuju basis (2x2) dengan cara ekspansi kofaktor.

2.2.1.1.2. ERO

Melakukan perhitungan determinan dengan membuat matriks eselon terlebih dahulu, lalu mengalikan seluruh diagonal matriks dan hasil perubahan determinan dari OBE yang menghasilkan determinan.

2.3. Modul InversSolver (InversSolver.java)

2.3.1. Class InversSolver

Merupakan class untuk menghitung matriks invers (jika ada) dari matriks masukan.

2.3.1.1. Methods

2.3.1.1.1. CofactorMatrix

Menerima matriks persegi dan mengembalikan invers matriks tersebut (jika ada) dengan metode kofaktor. Metode ini dibantu oleh metode CofactorExpansion dari DeterminantSolver.

2.3.1.1.2. GaussJordanMethod

Menerima matriks persegi dan mengembalikan invers matriks tersebut (jika ada) dengan metode Gauss-Jordan.

2.3.1.1.3. AdjointMethod

Menerima matriks persegi dan mengembalikan invers matriks tersebut (jika ada) dengan metode matriks adjoin. Matriks adjoin didapatkan dari

3. Package Problem

3.1. Modul Interpolation (Interpolation.java)

3.1.1. Class Interpolation

Merupakan class untuk menyelesaikan masalah interpolasi polinomial derajat n dengan n+1 titik.

3.1.1.1. Methods

3.1.1.1.1. addPoints

Menambah titik yang akan dikalkulasi interpolasi polinomialnya dalam bentuk Matrix.

3.1.1.1.2. fit

Melakukan kalkulasi terhadap n+1 titik yang telah ditambahkan sehingga menghasilkan koefisien persamaan interpolasi polinomial derajat n. Interpolasi hanya bisa di-fit jika terdapat solusi dari koefisien persamaan interpolasi atau terdapat cukup titik (lebih dari 1).

3.1.1.1.3. evaluate

Mengembalikan nilai evaluasi dari persamaan interpolasi terhadap nilai x pada argumen fungsi. Jika persamaan belum di fit akan memanggil fungsi fit, namun jika tidak berhasil untuk di-fit, mengembalikan nilai Double.MIN_VALUE (2^-1074).

3.1.1.2. Atribut

3.1.1.2.1. fit (isFit)

Untuk menentukan apakah *instance* berhasil di-fit.

3.1.1.2.2. count (getCount)

Untuk menyimpan nilai banyaknya point yang terdapat pada *instance*.

3.2. Modul Regression (Regression.java)

3.2.1. Class Regression

Merupakan class untuk menyelesaikan masalah regresi linear berganda.

3.2.1.1. Methods

3.2.1.1.1. getIntercept

Mengembalikan nilai *intercept* pada regresi yang telah di-fit.

3.2.1.1.2. getCoef

Mengembalikan nilai koefisien beta pada regresi yang di-fit.

3.2.1.1.3. hasSolution

Mengecek apakah fit pada regresi telah dilakukan.

3.2.1.1.4. fit

Melakukan fit pada matriks untuk mendapatkan nilai *intercept* dan koefisien.

3.2.1.1.5. predict

Melakukan prediksi pada array input dari nilai *intercept* dan beta, serta mengembalikan nilai prediksi.

3.2.1.2. Atribut

3.2.1.2.1. intercept (double)

Nilai beta ke-0.

3.2.1.2.2. coef (double[])

Merupakan array berisi beta ke-1 hingga beta ke-n.

3.2.1.2.3. hasSolution (boolean)

Untuk menandakan apakah *instance* sudah di-fit dengan suatu matriks.

4. Package I/O

4.1. Modul Parser (Parser.java)

4.1.1. Class Parser

Merupakan class pembantu untuk merubah string input menjadi variable dan merubah variable menjadi string output.

4.1.1.1. Methods

4.1.1.1.1. stringToMatrix

Merubah input string menjadi object matriks sesuai dengan jumlah baris dan kolomnya.

4.1.1.1.2. fileToMatrix

Menerima input file yang diubah menjadi string lalu menjadi object matriks sesuai dengan baris dan kolomnya.

4.1.1.1.3. stringToFile

Menerima variabel (biasanya output solver) yang akan dituliskan menjadi file.

4.1.1.1.4. coefRegressionToString

Menerima variabel koefisien yang dihasilkan class regression dan menuliskannya dalam bentuk persamaan untuk di output.

4.1.1.1.5. stringToDoubleArray

Merubah input string yang dipisahkan dengan spasi menjadi array of double.

5. Package UI

5.1. Modul Main (Main.java)

5.1.1. Class Main

Merupakan class utama dalam program yang berisikan UI (auto generated menggunakan java swing).

5.1.1.1. Atribut

5.1.1.1.1. outputFormat (Util.Formatting)

Berisikan setting formatting yang dipilih oleh user melewati GUI.

BAB 4
Eksperimen

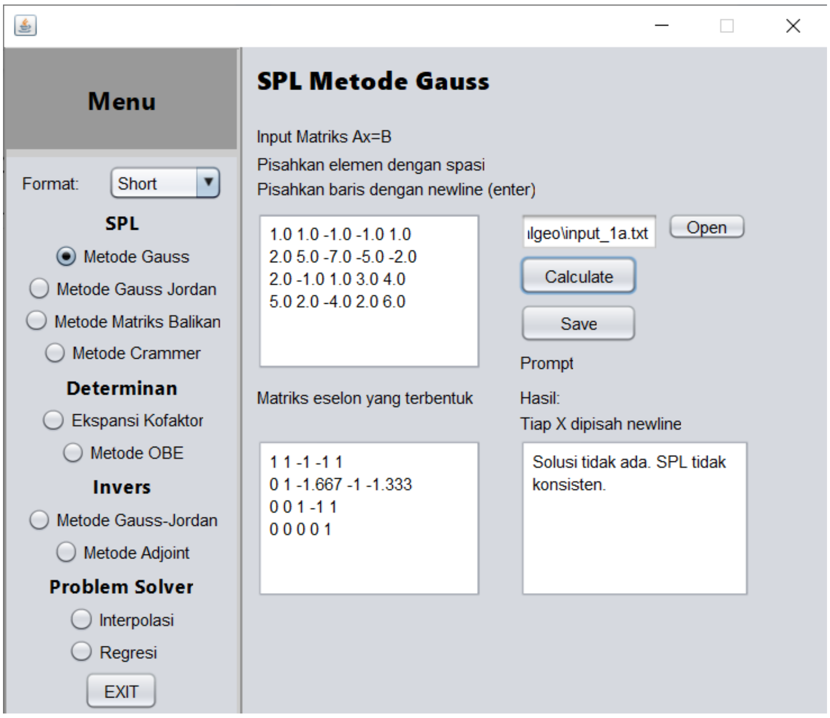
Studi Kasus No.1

Temukan solusi SPL *Ax=b*, berikut:

Bagian a

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

Gambar 4.1.1 Matriks Ax=b pada nomor 1a

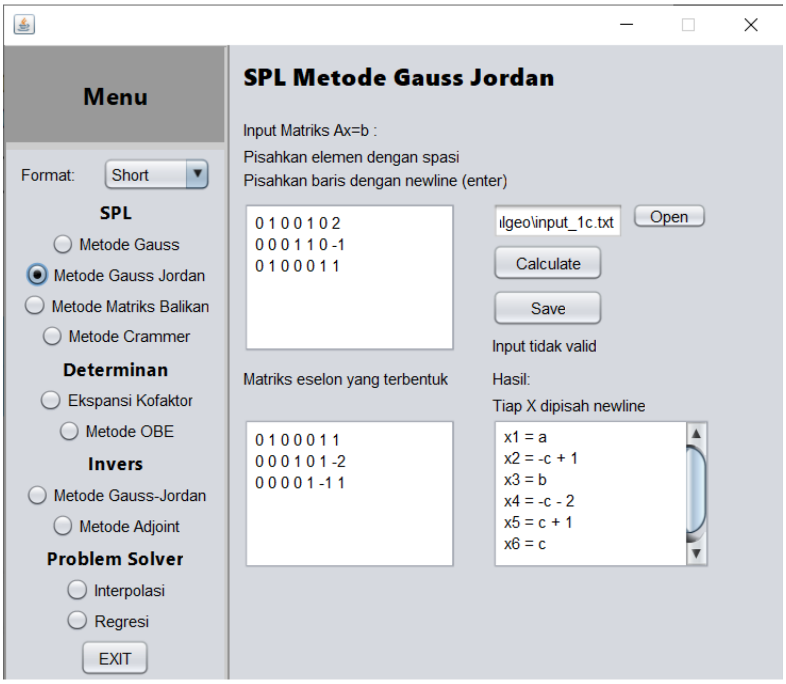


Gambar 4.1.2 Solusi untuk matriks pada nomor 1a

Bagian b

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

Gambar 4.1.3 Matriks Ax=b pada nomor 1b

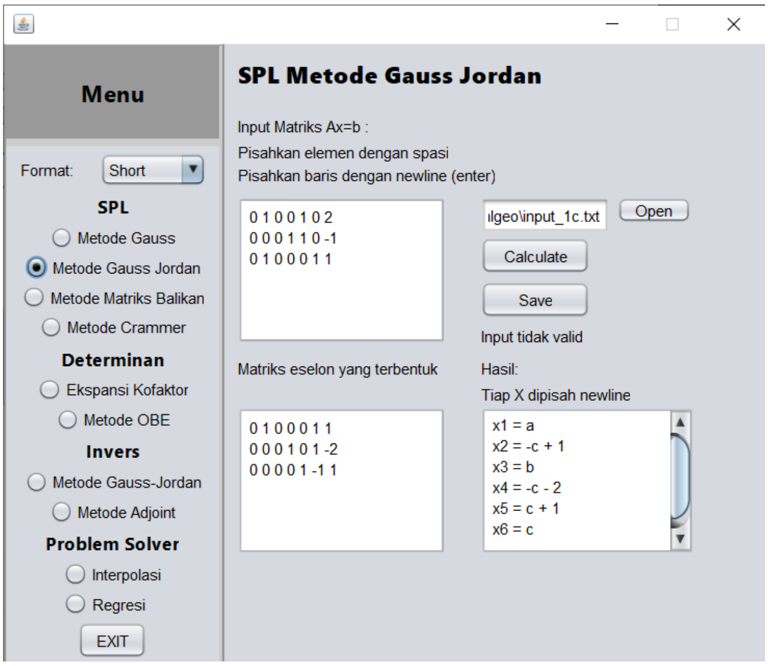


Gambar 4.1.4 Solusi untuk matriks pada nomor 1b

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

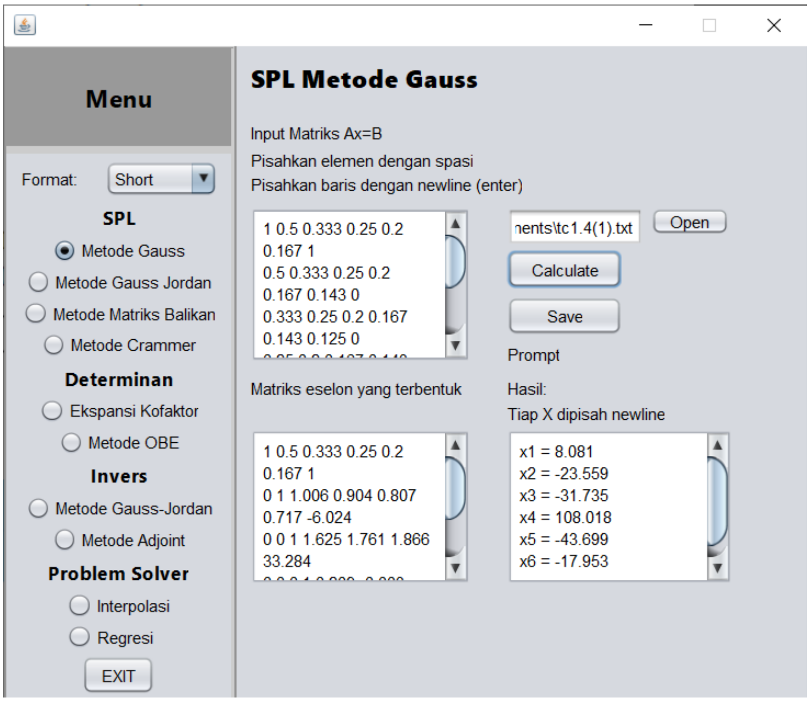
Gambar 4.1.5 Matriks Ax=b pada nomor 1b

Bagian c

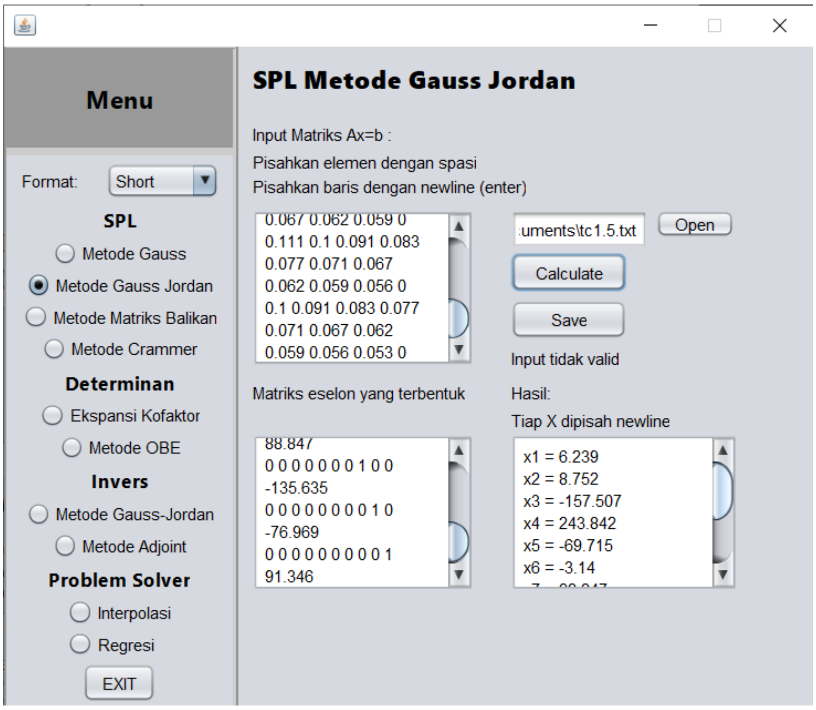


Gambar 4.1.6 Solusi untuk matriks pada nomor 1c

Bagian d



Gambar 4.1.7 Solusi untuk matriks pada nomor 1d, yaitu matriks Hilbert dengan n=6



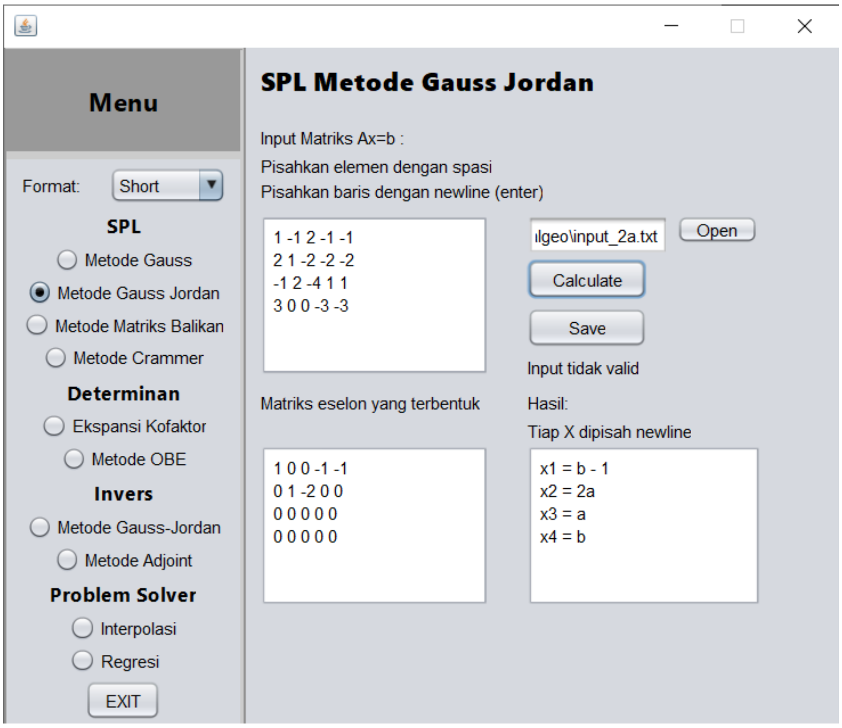
Gambar 4.1.8 Solusi untuk matriks pada nomor 1d, yaitu matriks Hilbert dengan n=10

Studi Kasus No.2

Bagian a

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

Gambar 4.2.1 Matriiks Augmented pada nomor 2a

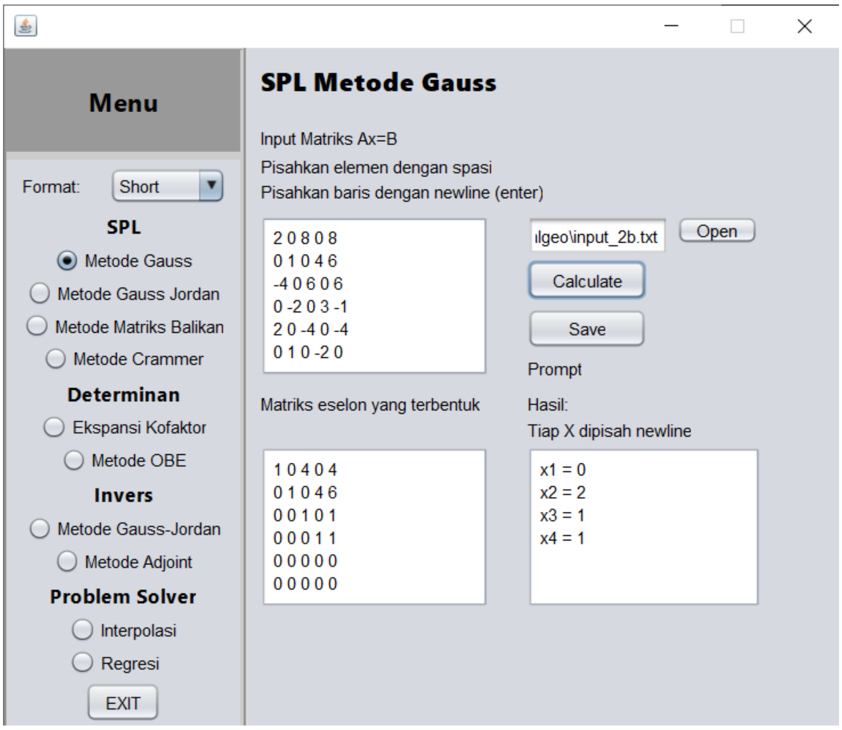


Gambar 4.2.2 Solusi untuk matriks augmented pada nomor 2a

Bagian b

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

Gambar 4.2.3 Matriiks Augmented pada nomor 2b



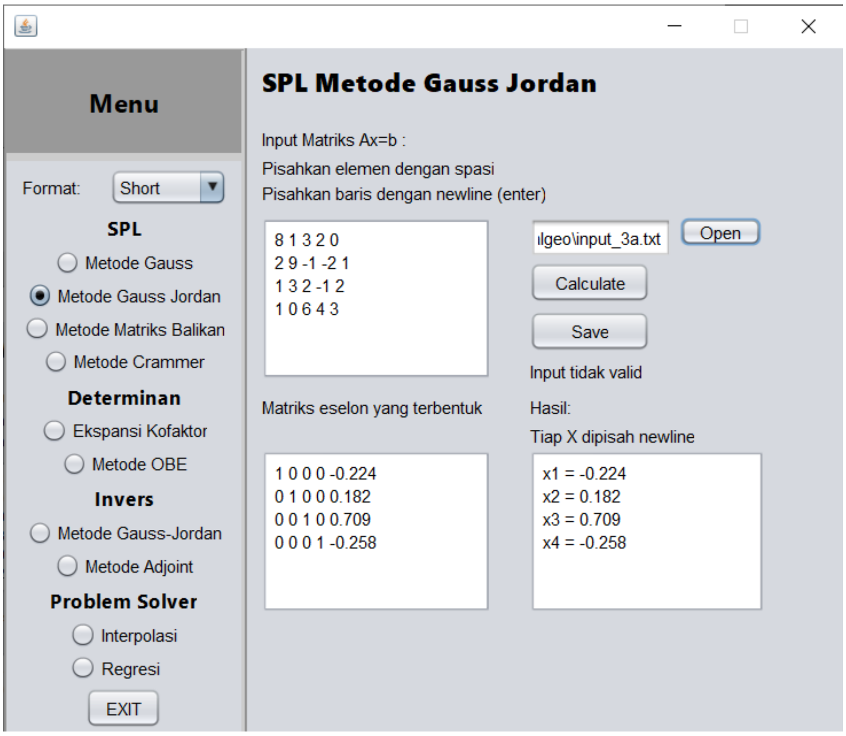
Gambar 4.2. Solusi untuk matriks augmented pada nomor 2b

Studi Kasus No.3

Bagian a

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

Gambar 4.3.1 SPL pada nomor 3a

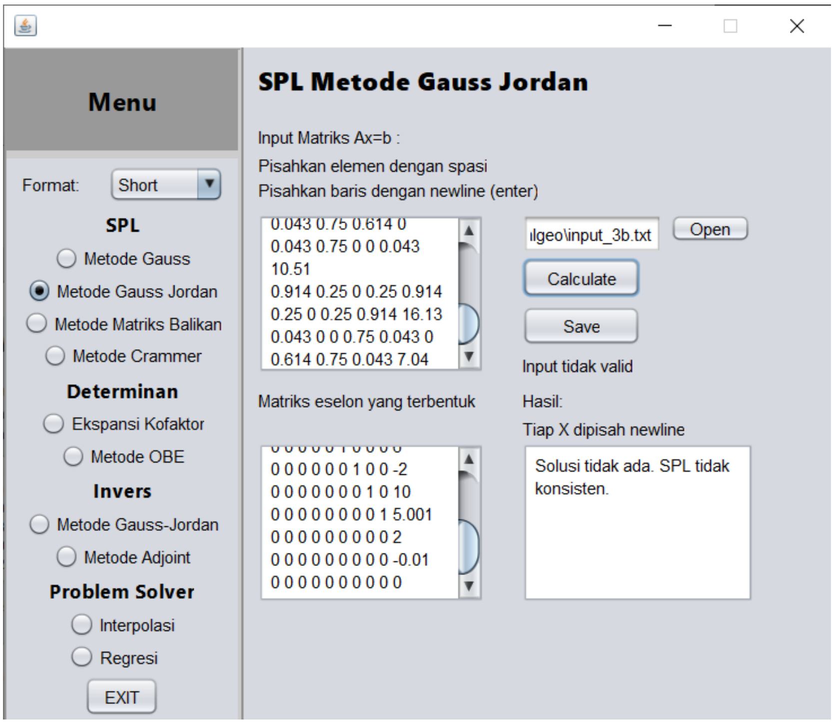


Gambar 4.3.2 Solusi untuk matriks dalam SPL pada nomor 3a

Bagian b

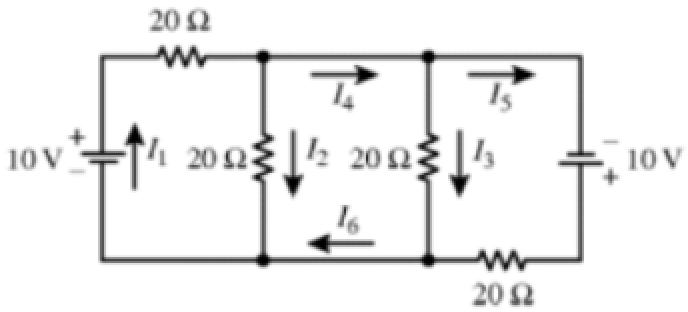
$$
\begin{aligned}
x_7 + x_8 + x_9 &= 13.00 \\
x_4 + x_5 + x_6 &= 15.00 \\
x_1 + x_2 + x_3 &= 8.00 \\
0.04289(x_3 + x_5 + x_7) + 0.75(x_6 + x_8) + 0.61396x_9 &= 14.79 \\
0.91421(x_3 + x_5 + x_7) + 0.25(x_2 + x_4 + x_6 + x_8) &= 14.31 \\
0.04289(x_3 + x_5 + x_7) + 0.75(x_2 + x_4) + 0.61396x_1 &= 3.81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

Gambar 4.3.3 SPL pada nomor 3b

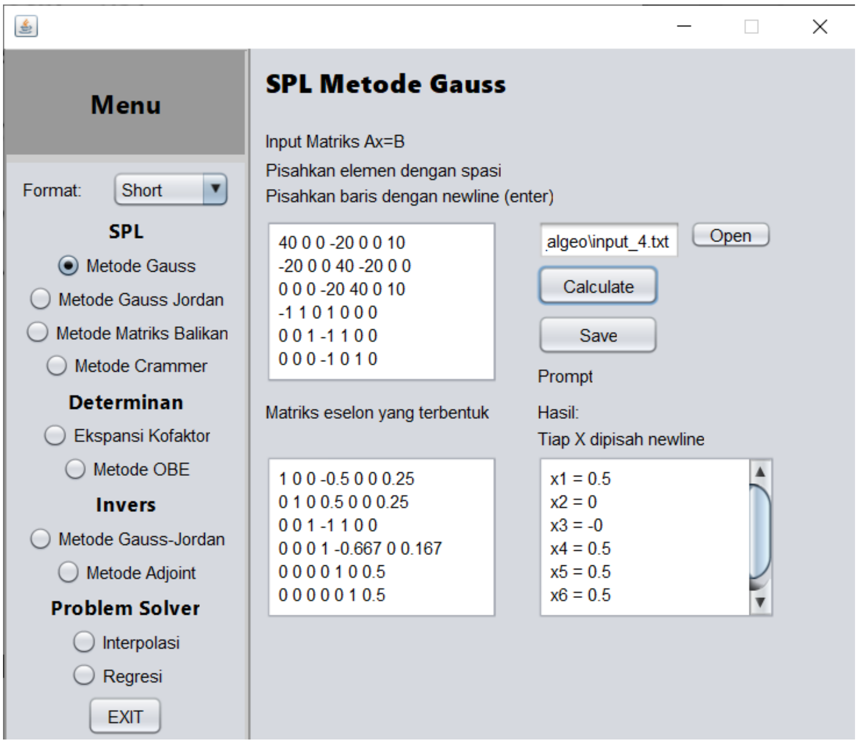


Gambar 4.3. Solusi untuk matriks dalam SPL pada nomor 3b

Studi Kasus No.4

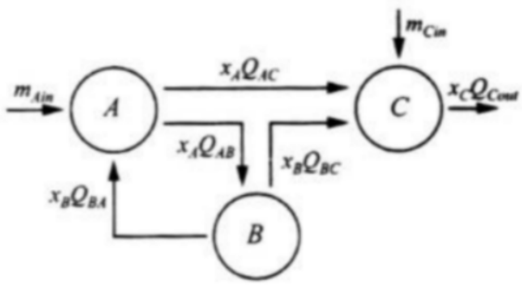


Gambar 4.4.1 Rangkaian listrik pada nomor 4



Gambar 4.4. Solusi untuk arus dalam persamaan rangkaian listrik pada nomor 4

Studi Kasus No.5



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

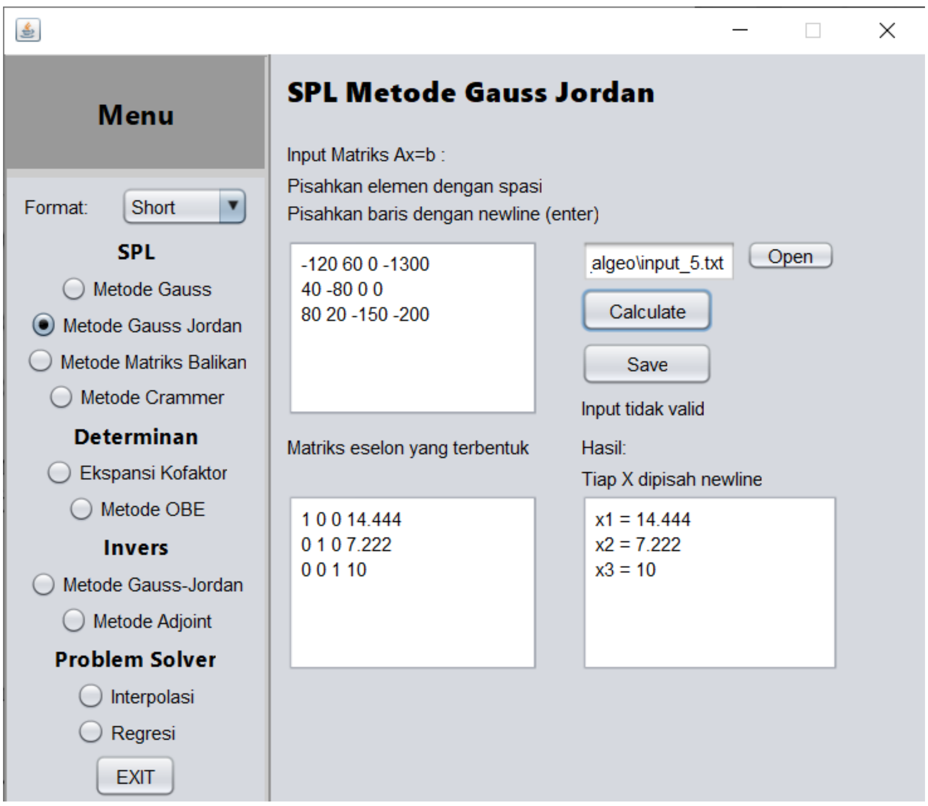
$$\text{A:}\quad m_{A_{in}} + Q_{BA}x_B - Q_{AB}x_A - Q_{AC}x_A = 0$$

$$\text{B:}\quad Q_{AB}x_A - Q_{BA}x_B - Q_{BC}x_B = 0$$

$$\text{C:}\quad m_{C_{in}} + Q_{AC}x_A + Q_{BC}x_B - Q_{C_{out}}x_C = 0$$

Tentukan solusi x_A, x_B, x_C dengan menggunakan parameter berikut : $Q_{AB} = 40$, $Q_{AC} = 80$, $Q_{BA} = 60$, $Q_{BC} = 20$ dan $Q_{Cout} = 150$ m^3/s dan $m_{Ain} = 1300$ dan $m_{Cin} = 200$ mg/s.

Gambar 4.5.1 Penjelasan soal reaktor pada nomor 5



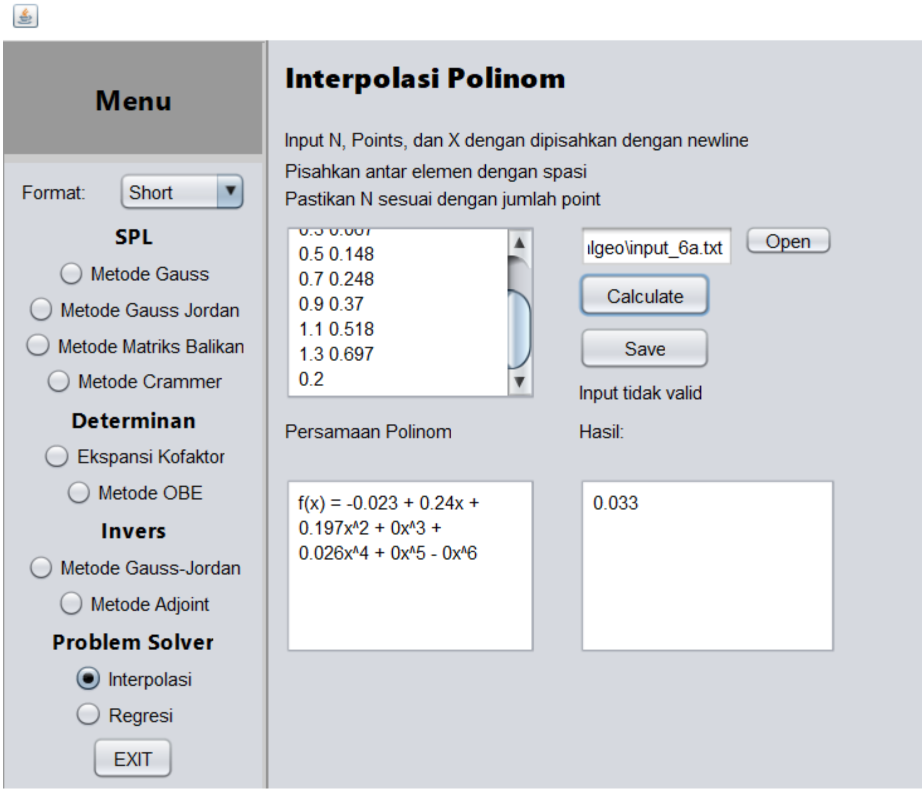
Gambar 4.5. Solusi untuk permasalahan sistem reaktor dalam SPL pada nomor 5

Studi Kasus No.6

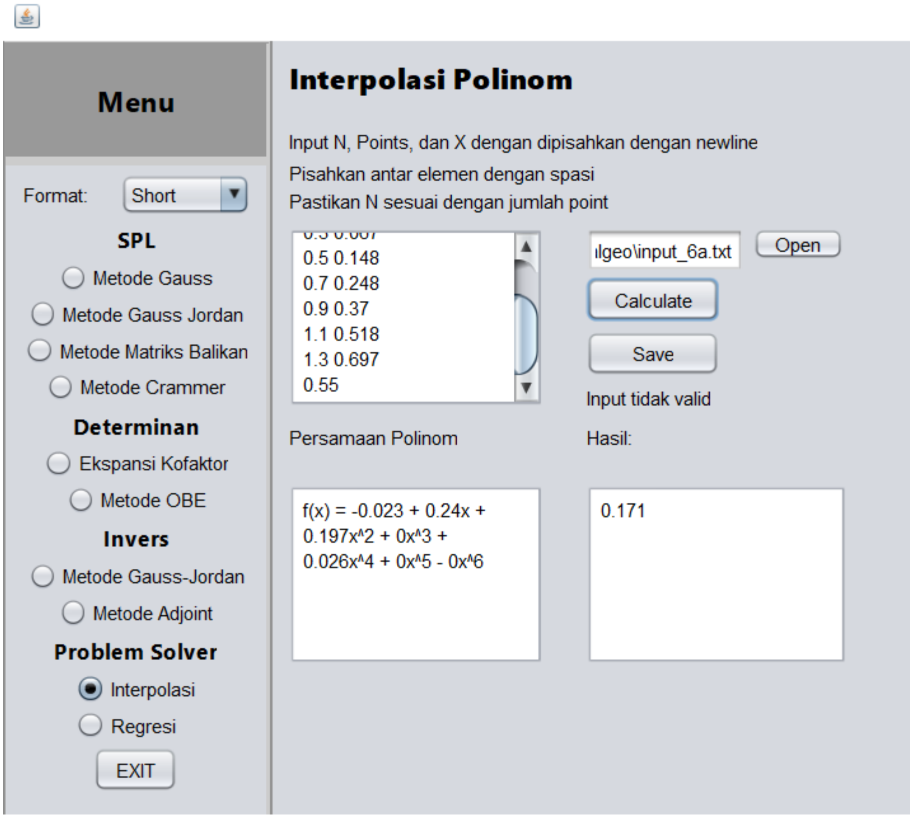
Bagian a

x	0.1	0.3	0.5	0.7	0.9	1.1	1.3
f(*x*)	0.003	0.067	0. 148	0.248	0.370	0.518	0.697

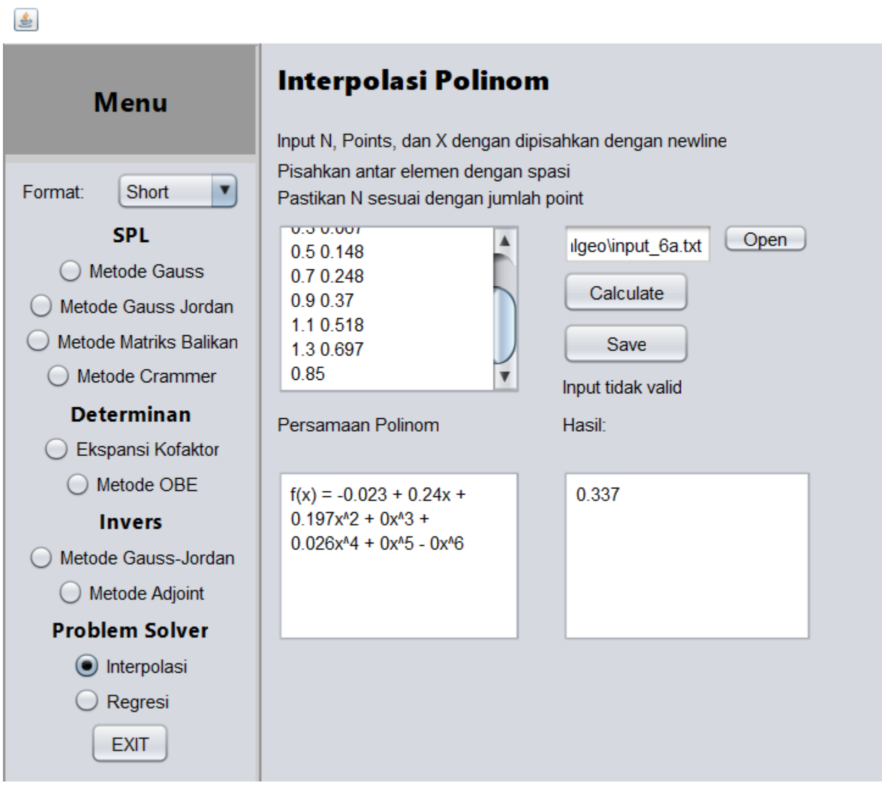
Gambar 4.6.1 Nilai x dan fungsi f(x) pada nomor 6a



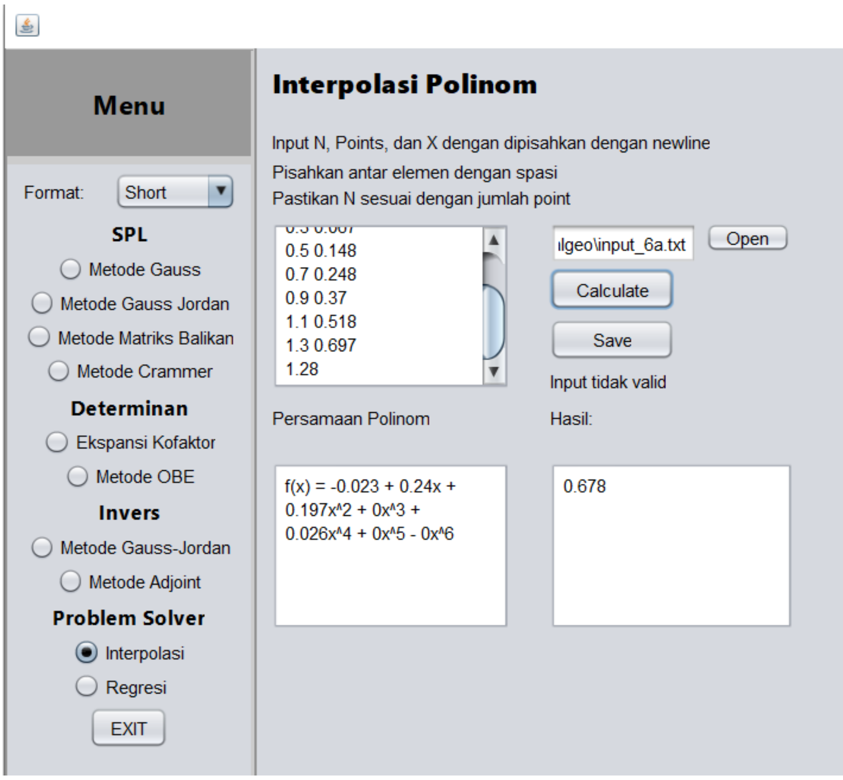
Gambar 4.6.2 Solusi untuk interpolasi polinom pada nomor 6a dengan x = 0.2



Gambar 4.6.3 Solusi untuk interpolasi polinom pada nomor 6a dengan x = 0.55



Gambar 4.6.4 Solusi untuk interpolasi polinom pada nomor 6a dengan x = 0.85

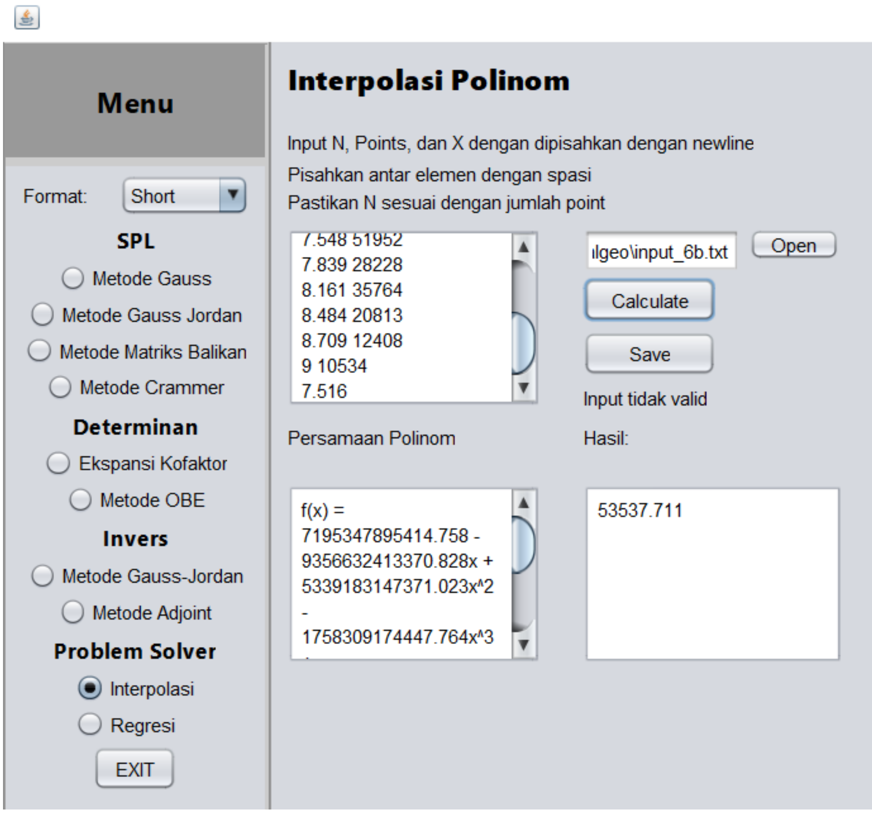


Gambar 4.6.5 Solusi untuk interpolasi polinom pada nomor 6a dengan x = 1.28

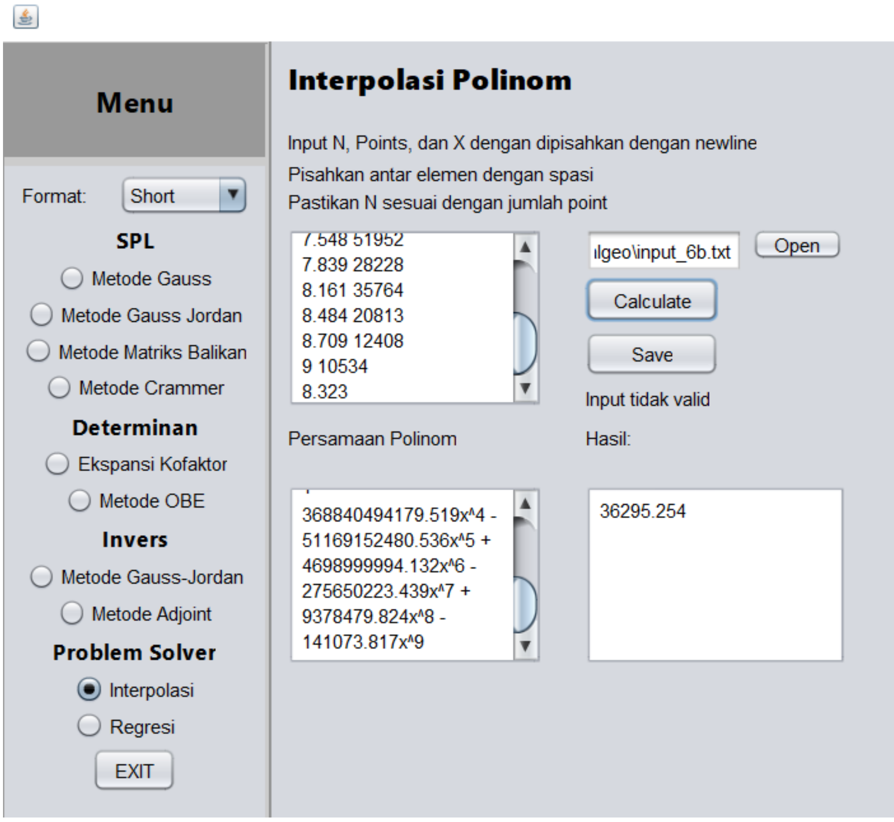
Bagian b

Tanggal	Tanggal (desimal)	Jumlah Kasus Baru
17/06/2021	6,567	12.624
30/06/2021	7	21.807
08/07/2021	7,258	38.391
14/07/2021	7,451	54.517
17/07/2021	7,548	51.952
26/07/2021	7,839	28.228
05/08/2021	8,161	35.764
15/08/2021	8,484	20.813
22/08/2021	8,709	12.408
31/08/2021	9	10.534

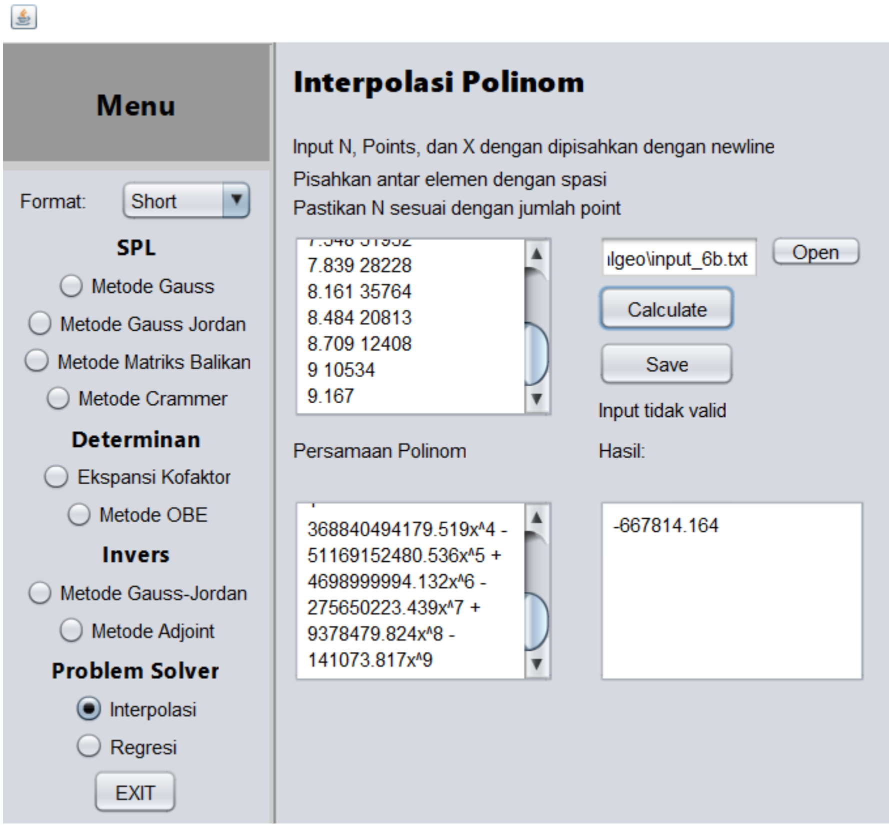
Gambar 4.6.6 Data Kasus Covid pada nomor 6b



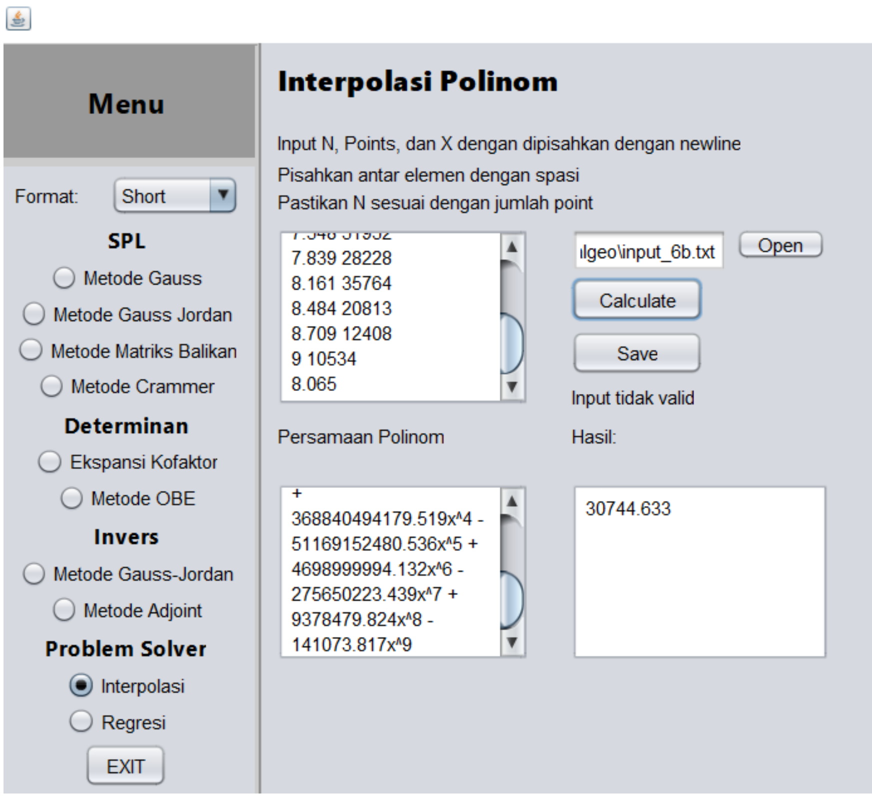
Gambar 4.6.7 Solusi untuk interpolasi polinom untuk kasus covid pada nomor 6b untuk tanggal 16/07/2021



Gambar 4.6.8 Solusi untuk interpolasi polinom untuk kasus covid pada nomor 6b untuk tanggal 10/08/2021



Gambar 4.6.9 Solusi untuk interpolasi polinom untuk kasus covid pada nomor 6b untuk tanggal 05/09/2021

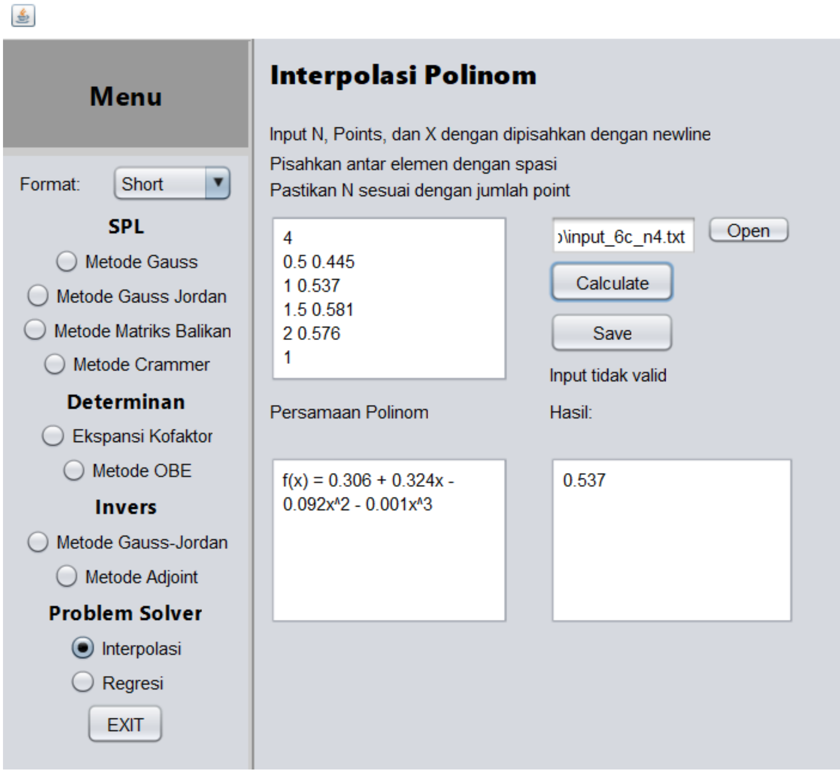


Gambar 4.6.10 Solusi untuk interpolasi polinom untuk kasus covid pada nomor 6b untuk tanggal 02/08/2021

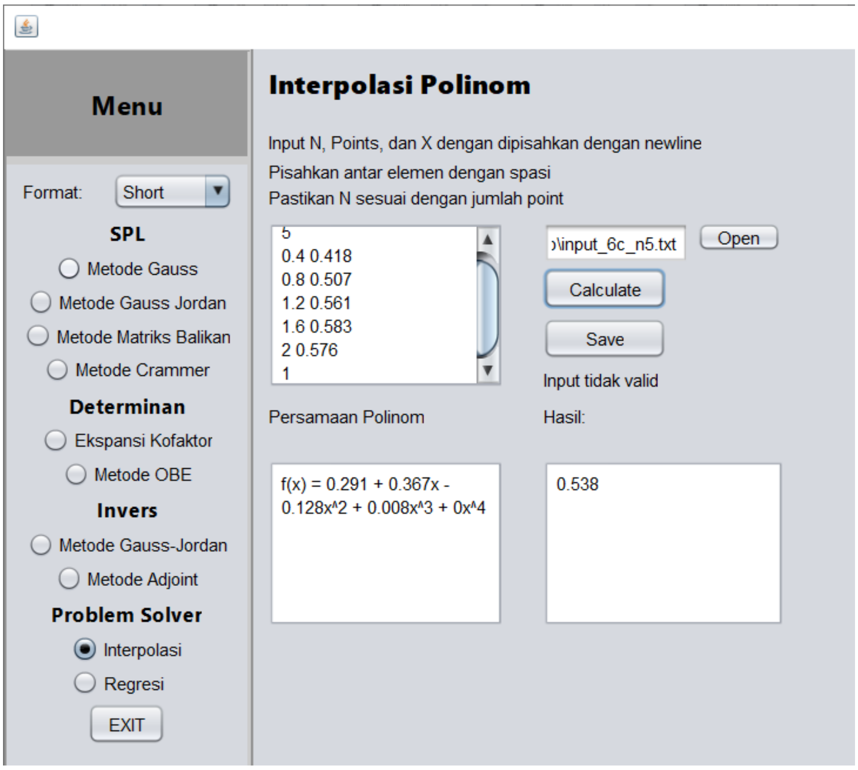
Bagian c

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

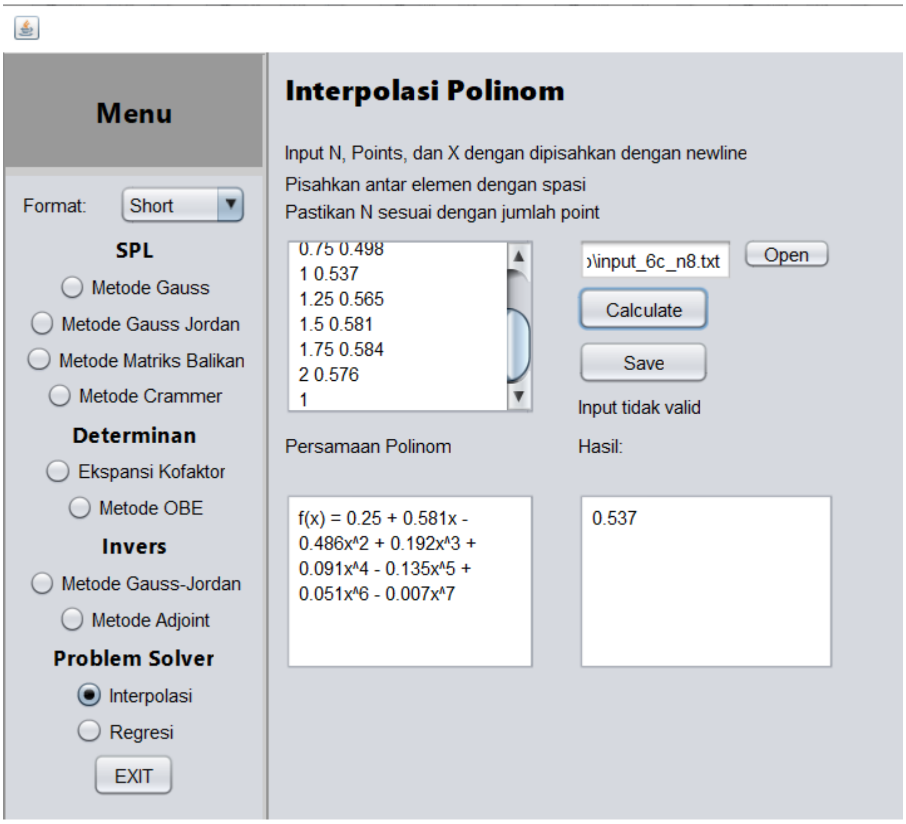
Gambar 4.6.11 Persamaan yang akan disederhanakan pada nomor 6c



Gambar 4.6.12 Solusi untuk penyederhanaan fungsi dengan interpolasi polinom pada nomor 6c dengan n=4



Gambar 4.6.13 Solusi untuk penyederhanaan fungsi dengan interpolasi polinom pada nomor 6c dengan n=5



Gambar 4.6.14 Solusi untuk penyederhanaan fungsi dengan interpolasi polinom pada nomor 6c dengan n=8

Studi Kasus No.7

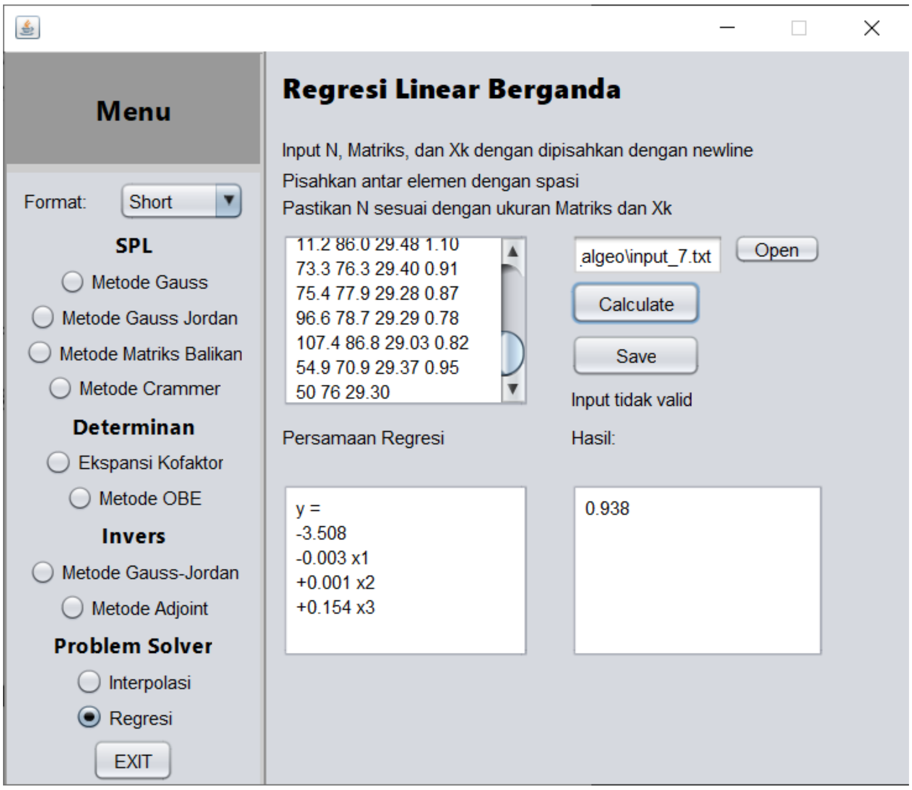
$$20b_0 + 863.1b_1 + 1530.4b_2 + 587.84b_3 = 19.42$$

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

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

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

Gambar 4.7.1 Persamaan Multiple Linear Regression pada nomor 7



Gambar 4.7.2 Solusi untuk regresi linear berganda pada nomor 7

BAB 5
Kesimpulan

Kesimpulan

Dalam pengerjaan tugas besar ini, kami mengimplementasi materi kuliah Aljabar Linier dan Geometri IF2123 dalam bentuk modul yang dikemas menjadi program dalam bahasa Java. Kami berhasil memenuhi spek tugas besar yang disampaikan dan menambahkan beberapa fitur tambahan.

Program yang kami susun mempunyai banyak fitur perhitungan seperti; menghitung Sistem Persamaan Linear menggunakan metode Gauss, Gauss-Jordan, Matriks Balikan, dan Cramer; menghitung determinan matriks menggunakan metode Ekspansi Kofaktor dan Operasi Baris Elementer; menghitung matriks invers menggunakan metode Gauss-Jordan dan Adjoint. Selain itu, program kami juga mempunyai fitur untuk melakukan interpolasi dan regresi dibantu dengan fitur perhitungannya.

Kami menyajikan program kami dengan GUI sehingga mudah diakses dan digunakan. Pengguna juga memberikan input melalui file dan output melalui file jika tidak ingin memasukkan secara manual. Kami juga memberi pilihan formatting agar lebih mudah dibaca oleh pengguna sesuai kebutuhannya.

Saran

Dari kelompok kami, ada beberapa fitur yang seharusnya bisa dikembangkan, tetapi dikarenakan adanya *time-constraint* sehingga kami memutuskan untuk tidak mengembangkannya. Fitur tersebut seperti:

- Pembentuk matriks augmented dari persamaan SPL
- Visualisasi/penggambaran proses pembentukan matriks eselon menggunakan OBE
- Penulisan persamaan menggunakan LaTeX
- Grafik interpolasi dan grafik regresi
- Batch input/output file txt

Refleksi

a. 13520062 Rifqi Naufal Abdjul

   Saya belajar banyak hal terkait kebersihan kode dan cara membuat project dengan baik dan benar dari teman saya. Saya juga belajar banyak tentang menyusun UI dan OOP dan berusaha mengimplementasikannya dalam tugas besar ini walau masih kurang sempurna.

b. 13520103 Amar Fadil

   Dari tugas besar ini, saya belajar untuk mengorganisasi direktori project untuk Java. Saya juga belajar mengenai unit testing dan OOP varian java, serta melatih best practice seperti DRY pada project ini.

c. 13520153 Vito Ghifari

   Pada tugas besar ini, saya menjadi lebih memahami tentang penyelesaian menggunakan Gauss dan Gauss-Jordan. Karena sebelumnya saya *stuck* cukup lama untuk menerjemahkan ke dalam kode untuk solusi parametrik. Saya juga menjadi paham tentang regresi dalam bentuk matriks serta penyelesaiannya. Karena tugas besar ini juga, saya *terjun* sedikit ke Java dan OOP.

Daftar Pustaka

Howard Anton, Dkk. *Elementary Linier Algebra with Applications ninth edition*. USA: John Wiley & Jons, 2005.

Cosentino, Scott. *Programming Basic Java GUIs With Swing.*
Diakses dari https://scottc130.medium.com/programming-basic-java-guis-with-swing-854dfa6decab

Dhalla, Adam. *The Normal Equation for Linear Regression*. Artificial Intelligence in Plain English. Diakses dari https://ai.plainenglish.io/the-normal-equation-for-linear-regression-25fddea63899

PennState Eberly College of Science. *5.3 - The Multiple Linear Regression Model*. STAT 462. Diakses dari https://online.stat.psu.edu/stat462/node/131/