LAPORAN TUGAS BESAR I

IF2123 ALJABAR LINIER DAN GEOMETRI

Laporan ini dibuat untuk memenuhi tugas

Mata Kuliah IF 2123 Aljabar Linier dan Geometri

Disusun Oleh:

Kelompok "Dispenser"

Raden Rafly Hanggaraksa Budiarto (13522014)

Rayendra Althaf Taraka Noor (13522107)

Rayhan Fadhlan Azka (13522095)

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

SEMESTER I TAHUN 2022/2023

Daftar Isi

BAB 1
DESKRIPSI MASALAH

Membuat program dalam Bahasa Java untuk:

1. Menentukan sistem persamaan linear dengan matriks menggunakan metode eliminasi gauss, eliminasi gauss-jordan, matriks balikan, atau cramer.
2. Menentukan determinan dari matriks dengan menggunkan metode upper triangle atau kofaktor.
3. Menafsirkan nilai suatu angka dari beberapa titik menggunakan interpolasi polinom.
4. Mengaproksimasi fungsi di antara titik – titik yang diketahui menggunkan metode *Bicubinc Spine Interpolation*.
5. Memprediksi nilai menggunakan regresi linear.

Spesifikasi dari program yang dibuat adalah:

1. Program dapat menerima masukan (input) baik dari keyboard maupun membaca masukan dari file text. Untuk SPL, masukan dari keyboard adalah m, n, koefisien aij , dan bi. 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 aij. Masukan dari file berbentuk matriks, setiap elemen matriks dipisah oleh spasi. Misalnya,

   3 4.5 2.8

   -3 7 8.3

   0.5 -10 -9

   Luaran (output) disesuaikan dengan persoalan (determinan atau invers) dan penghitungan balikan/invers dilakukan dengan metode matriks balikan dan adjoin.

3. Untuk persoalan interpolasi, masukannya jika dari keyboard adalah n, (x0, y0), (x1, y1), …, (xn, yn), dan nilai x yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung. Masukan kemudian dilanjutkan dengan satu buah baris berisi satu buah nilai x yang akan ditaksir menggunakan fungsi interpolasi yang telah didefinisikan. Misalnya jika titik-titik datanya adalah (8.0, 2.0794), (9.0, 2.1972), dan (9.5, 2.2513) dan akan mencari nilai y saat x = 8.3, maka di dalam file text ditulis sebagai berikut:

   ```
   8.0 2.0794
   ```

```
9.0 2.1972
9.5 2.2513
8.3
```

4. Untuk persoalan regresi, masukannya jika dari keyboard adalah n (jumlah peubah x), m (jumlah sampel), semua nilai-nilai x1i, x2i, …, xni, nilai yi, dan nilai-nilai xk yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung.
5. Untuk persoalan SPL, luaran 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 x4 = -2, x3 = 2s – t, x2 = s, dan x1 = t).
6. Untuk persoalan polinom interpolasi dan regresi, luarannya adalah persamaan polinom/regresi dan taksiran nilai fungsi pada *x* yang diberikan. Contoh luaran untuk interpolasi adalah

$$f(x) = -0.0064x^2 + 0.2266x + 0.6762, \quad f(5) = \ldots$$

dan untuk regresi adalah

$$f(x) = -9.5872 + 1.0732x_1, \quad f(x_k) = \ldots$$

7. Untuk persoalan bicubic spline interpolation, masukan dari file text (.txt) yang berisi matriks berukuran 4 x 4 yang berisi konfigurasi nilai fungsi dan turunan berarah disekitarnya, diikuti dengan nilai a dan b untuk mencari nilai f(a, b).

Misalnya jika nilai dari f(0, 0), f(1, 0), f(0, 1), f(1, 1), fx(0, 0), fx(1, 0), fx(0, 1), fx(1, 1), fy(0, 0), fy(1, 0), fy(0, 1), fy(1, 1), fxy(0, 0), fxy(1, 0), fxy(0, 1), fxy(1, 1) berturut-turut adalah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 serta nilai a dan b yang dicari berturut-turut adalah 0.5 dan 0.5 maka isi file text ditulis sebagai berikut:

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

Luaran yang dihasilkan adalah nilai dari f(0.5, 0.5).

8. Luaran program harus dapat ditampilkan pada layar komputer dan dapat disimpan ke dalam file.
9. Bahasa program yang digunakan adalah Java. Anda bebas untuk menggunakan versi java apapun dengan catatan di atas java versi 8 (8/9/11/15/17/19/20).

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

Sistem Persamaaan Linier

Determinan

Matriks balikan

Interpolasi Polinom

Interpolasi Bicubic Spline

Regresi linier berganda

Keluar
```

    Untuk pilihan menu nomor 1 ada sub-menu lagi yaitu pilihan metode:

```
Metode eliminasi Gauss

Metode eliminasi Gauss-Jordan

Metode matriks balikan

Kaidah Cramer
```

    Begitu juga untuk pilihan menu nomor 2 dan 3.

BAB II
TEORI SINGKAT

A. Abstraksi

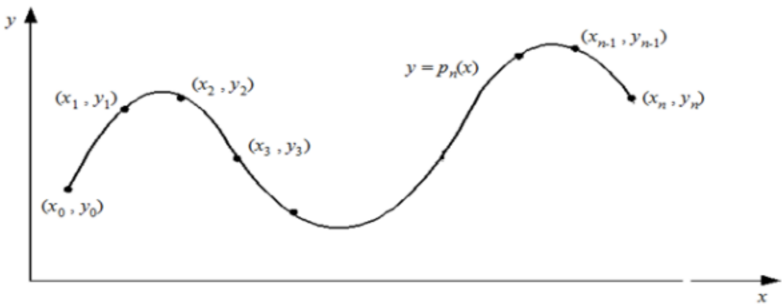
Sistem persamaan linier (SPL) banyak ditemukan di dalam bidang sains dan rekayasa. Anda sudah mempelajari berbagai metode untuk menyelesaikan SPL, termasuk menghitung determinan matriks. Sembarang SPL dapat diselesaikan dengan beberapa metode, yaitu metode eliminasi Gauss, metode eliminasi Gauss-Jordan,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

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

Gambar 1 Eliminasi Gauss dilakukan dengan matriks eselon baris dan eliminasi Gauss-Jordan dengan matriks eselon baris tereduksi.

B. Interpolasi Polinomial

Persoalan interpolasi polinom adalah sebagai berikut: Diberikan n+1 buah titik berbeda, $(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$. Tentukan polinom $p_n(x)$ yang menginterpolasi (melewati) semua titik-titik tersebut sedemikian rupa sehingga $y_i = p_n(x_i)$ untuk $i = 0, 1, 2, \ldots, n$.



Gambar 2 Ilustrasi beberapa titik yang diinterpolasi secara polinomial.

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 + a2x2 + … + anxn. 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 + a2x2 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 + a2x2 + a3x3, 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 + a2x2 + … + anxn untuk i = 0, 1, 2, …, n, akan diperoleh n buah sistem persamaan lanjar dalam a0, a1, a2, …, an,

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

Solusi sistem persamaan lanjar ini, yaitu nilai a0, a1, …, an, 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 p2(x) = a0 + a1x + a2x2. 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 a0 = 0.6762,   a1 = 0.2266, dan a2 = -0.0064. Polinom interpolasi yang melalui ketiga buah titik tersebut adalah p2(x) = 0.6762 + 0.2266x - 0.0064x2. 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.

C. Regresi Linear 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 persamaan jadi untuk menghitung regresi linear sederhana, terdapat persamaan 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:

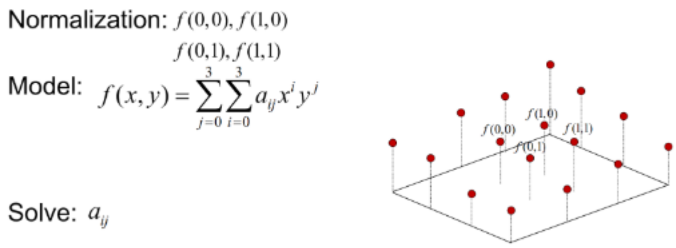
$$
\begin{array}{lllll}
nb_0 + b_1 \sum_{i=1}^{n} x_{1i} & + b_2 \sum_{i=1}^{n} x_{2i} & + \cdots & + b_k \sum_{i=1}^{n} x_{ki} & = \sum_{i=1}^{n} y_i \\
b_0 \sum_{i=1}^{n} x_{1i} + b_1 \sum_{i=1}^{n} x_{1i}^2 & + b_2 \sum_{i=1}^{n} x_{1i}x_{2i} & + \cdots & + b_k \sum_{i=1}^{n} x_{1i}x_{ki} & = \sum_{i=1}^{n}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

Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

D. Bicubic Spline Interpolation

Bicubic spline interpolation adalah metode interpolasi yang digunakan untuk mengaproksimasi fungsi di antara titik-titik data yang diketahui. *Bicubic spline interpolation* melibatkan konsep *spline* dan konstruksi serangkaian polinomial kubik di dalam setiap sel segi empat dari data yang diberikan. Pendekatan ini menciptakan permukaan yang halus dan kontinu, memungkinkan untuk perluasan data secara visual yang lebih akurat daripada metode interpolasi linear.

Dalam pemrosesan menggunakan interpolasi *bicubic spline* digunakan 16 buah titik, 4 titik referensi utama di bagian pusat, dan 12 titik di sekitarnya sebagai aproksimasi turunan dari keempat titik referensi untuk membangun permukaan bikubik. Bentuk pemodelannya adalah sebagai berikut.



Gambar 3 Pemodelan interpolasi bicubic spline.

Selain melibatkan model dasar, juga digunakan model turunan berarah dari kedua sumbu, baik terhadap sumbu x, sumbu y, maupun keduanya. Persamaan polinomial yang digunakan adalah sebagai berikut.

$$f(x,y)=\sum_{i=0}^{3}\sum_{j=0}^{3}a_{ij}x^i y^j$$

$$f_x(x,y)=\sum_{j=0}^{3}\sum_{i=1}^{3}a_{ij}ix^{i-1}y^j$$

$$f_y(x,y)=\sum_{j=1}^{3}\sum_{i=0}^{3}a_{ij}jx^i y^{j-1}$$

$$f_{xy}(x,y)=\sum_{j=0}^{3}\sum_{i=0}^{3}a_{ij}ijx^{i-1}y^{j-1}$$

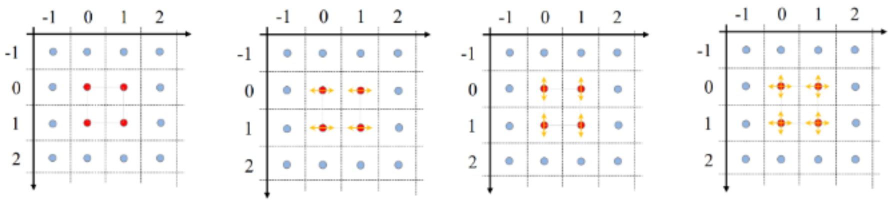
Dengan menggunakan nilai fungsi dan turunan berarah tersebut, dapat terbentuk sebuah matriks solusi X yang membentuk persamaan penyelesaian sebagai berikut.

$$y = Xa$$

$$\begin{bmatrix} f(0,0) \\ f(1,0) \\ f(0,1) \\ f(1,1) \\ f_x(0,0) \\ f_x(1,0) \\ f_x(0,1) \\ f_x(1,1) \\ f_y(0,0) \\ f_y(1,0) \\ f_y(0,1) \\ f_y(1,1) \\ f_{xy}(0,0) \\ f_{xy}(1,0) \\ f_{xy}(0,1) \\ f_{xy}(1,1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 2 & 4 & 6 & 0 & 3 & 6 & 9 \end{bmatrix} \begin{bmatrix} a_{00} \\ a_{10} \\ a_{20} \\ a_{30} \\ a_{01} \\ a_{11} \\ a_{21} \\ a_{31} \\ a_{02} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{03} \\ a_{13} \\ a_{23} \\ a_{33} \end{bmatrix}$$

Perlu diketahui bahwa elemen pada matriks X adalah nilai dari setiap komponen koefisien aij yang diperoleh dari persamaan fungsi maupun persamaan turunan yang telah dijelaskan sebelumnya. Sebagai contoh, elemen matriks X pada baris 8 kolom ke 2 adalah koefisien dari a10 pada ekspansi sigma untuk fx(1, 1) sehingga diperoleh nilai konstanta 1 x 11-1 x 10 = 1, sesuai dengan isi matriks X.

Nilai dari vektor a dapat dicari dari persamaan y = Xa, lalu vektor a tersebut digunakan sebagai nilai variabel dalam f(x, y), sehingga terbentuk fungsi interpolasi bicubic sesuai model. Tugas Anda pada studi kasus ini adalah membangun persamaan f(x, y) yang akan digunakan untuk melakukan interpolasi berdasarkan nilai f(a, b) dari masukan matriks 4 x 4. Nilai masukan a dan b berada dalam rentang [0, 1]. Nilai yang akan diinterpolasi dan turunan berarah di sekitarnya dapat diilustrasikan pada titik berwarna merah pada gambar di bawah.



Gambar 4 Nilai fungsi yang akan di interpolasi pada titik merah, turunan berarah terhadap sumbu x, terhadap sumbu y, dan keduanya (kiri ke kanan).

Seperti yang telah dijelaskan sebelumnya bahwa interpolasi bicubic spline dapat digunakan untuk menciptakan permukaan yang halus pada gambar. Oleh karena itu, selain persamaan dasar y = Xa yang telah dijabarkan, persamaan ini juga dapat menggunakan data sebuah citra untuk menciptakan kualitas gambar yang lebih baik. Misalkan I(x, y) merupakan nilai dari suatu citra gambar pada posisi (x, y), maka dapat digunakan persamaan nilai dan persamaan turunan berarah sebagai berikut.

$$f(x,y) = I(x,y)$$

$$f_x(x,y) = [I(x+1,y) - I(x-1,y)]/2$$

$$f_y(x,y) = [I(x,y+1) - I(x,y-1)]/2$$

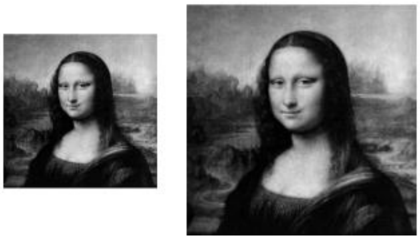
$$f_{xy}(x,y) = [I(x+1,y+1) - I(x-1,y) - I(x,y-1) - I(x,y)]/4$$

Sistem persamaan tersebut dapat dipetakan menjadi sebuah matriks (dalam hal ini matriks D) dengan gambaran lengkap seperti yang tertera di bawah.

$$y = DI$$

$$\begin{bmatrix} f(0,0) \\ f(1,0) \\ f(0,1) \\ f(1,1) \\ f_x(0,0) \\ f_x(1,0) \\ f_x(0,1) \\ f_x(1,1) \\ f_y(0,0) \\ f_y(1,0) \\ f_y(0,1) \\ f_y(1,1) \\ f_{xy}(0,0) \\ f_{xy}(1,0) \\ f_{xy}(0,1) \\ f_{xy}(1,1) \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I(-1,-1) \\ I(0,-1) \\ I(1,-1) \\ I(2,-1) \\ I(-1,0) \\ I(0,0) \\ I(1,0) \\ I(2,0) \\ I(-1,1) \\ I(0,1) \\ I(1,1) \\ I(2,1) \\ I(-1,2) \\ I(0,2) \\ I(1,2) \\ I(2,2) \end{bmatrix}$$

Dengan menggunakan kedua persamaan nilai y yang telah disebutkan dan dibahas sebelumnya, dapatkan nilai a yang lebih baik dan akurat dalam pemrosesan citra gambar, kemudian gunakan nilai dan persamaan f(x, y) yang terbentuk untuk memperbaiki kualitas citra gambar monokrom pasca pembesaran dengan skala tertentu dengan melakukan interpolasi bicubic spline. Berikut adalah contohnya.



Gambar 5 Sebuah citra gambar asal (kiri) dan hasil pembesarannya dengan skala 1.5 (kanan).

BAB III
IMPLEMENTASI PROGRAM

A. Matriks.java

Pada class ini, dideklarasikan atribut matriks berupa:

- double[][] matrix yang berperan sebagai kontainer.
- Row yang menyimpan panjang baris.
- Col yang menyimpan panjang kolom.
- Double tx dan ty digunakan untuk menyimpan titik yang ingin ditaksir

Method	Deskripsi
Matrix	Berperan sebagai konstruktor matrix. (Inisialisasi Matriks)
getRow	Mengembalikan nilai baris dari matrix.
getCol	Mengembalikan nilai kolom dari matrix.
setRow	Memberi nilai row dalam matrix.
setCol	Memberi nilai col dalam matrix.
initializeMatrix	Inisialisasi matriks.
setMatrix	Memberikan matriks nilai baris dan kolom.
setMatrixValue	Memberikan matriks nilai untuk baris dan kolom.
add	Menjumlah matriks dengan matriks lain.
perkalianMatrix	Mengalikan matriks dengan matriks lain.
swaprow	Menukar baris matriks.
copyMatrix	Memberi salinan dari suatu matriks
transpose	Transpos Matrix.
swapZeroToBottom	Menukar baris dengan nilai nol paling banyak ke bawah.
forwardOBE	Melakukan operasi baris elementer dengan langkah maju.
identity	Mengembalikan nilai matriks identitas NxN.
isRow0	Mengecek apakah nilai dari suatu baris bernilai 0
isRow0New	Mengecek apakah nilai dari suatu baris bernilai 0 dengan mengganggap kolom terakhir.
gauss	Melakukan operasi gauss terhadap matrix.
reverseOBE	Melakukan operasi baris elementer dengan langkah mundur.
isLeading1Present	Mengecek apakah nilai leading one ada pada suatu baris.

findLeading1	Mencari nilai leading one
gaussJordan	Melakukan operasi gauss jordan terhadap matrix.
pangkat	Mengembalikan nilai integer pangkat a^b.
printMatriks	Mencetak matriks ke konsol.
bacaMatriks	Membaca matriks dengan keyboard.
printInverseCofactor	Mencetak hasil inverse dari metode ekpansi kofaktor dan adjoin.
InverseMatrixIdentity	Mencetak hasil inverse dari metode gauss-jordan. (Identitas)
matrixBicubicSpline	Memanggil isiMatrixBicubic dan memvariasikan nilai a,b yang dipanggil dengan 0 dan 1;
isiMatrixBicubic	Digunakan oleh fungsi Matrix bicubic untuk mencari nilai f(a,b), fx(a,b), fy(a,b), dan fxy(a,b)
isSquare	Mengecek apakah matriks NxN.
buatFile	Menciptakan file (.txt) kedalam folder “dataoutput”.
matrixToString	Mengubah keluaran matriks menjadi string.
matrixToFile	Menuliskan hasil matriks kedalaam file.
writeStringToFile	Menuliskan input string kedalam file.
determinantGaussMatriks	Mencari deteriminan menggunakan metode upper triangle.
getMinor	Mencari minor dari matriks.
getAdjoin	Mencari adjoin dari matriks.
bacaFileMatrix	Membaca isi dari file (.txt).
bicMeasure	Menaksir nilai di titik yang ditentukan dengan persamaan bicubic yang sudah didapat.
getCramerSol	Mendapatkan solusi dengan SPL Cramer.
getDeterminantCofactor	Mencari determinan menggunakan metode kofaktor.
getSumOfLeading1	Mendapatkan jumlah seluruh leading 1 dalam matriks eselon.
countNonZeroRow	Menghitung jumlah baris yang tidak seluruhnya bernilai nol.
isSPLUnique	Memeriksa apakah persamaan memiliki solusi unik.
isRowInvalidSol	Mengecek apakah baris memenuhi syarat agar matriks tidak punya solusi
isSPLInfiniteSol	Memeriksa apakah persamaan memiliki solusi banyak.
isSPLInvalidValue	Memeriksa apakah persamaan tidak memiliki solusi.
printCramerSol	Mengubah solusi SPL cramer menjadi paragraph dalam bentuk string.

isMatrixCramerable	Mengecek apakah matriks dapat dilakukan operasi cramer
createNewRow	Membuat row baru untuk suatu matrix.
createArrayOfSol	Menyimpan solusi dalam bentuk array.
createArrayofParam	Mencipatkan array yang berisi alphabet.
isSolParam	Mengecek jika suatu baris dalam matriks memiliki parameter.
isMatrixAugmented	Mengecek apakah matrix augmented.
printSPLInfSol	Mencetak solusi SPL tak hingga kedalam string.
printSPLSol	Mencetak solusi SPL.
printParametricValue	Mencetak nilai parametrik.
getSPLGauss	Melakukan operasi sistem persamaan linear gauss.
removeCol	Menghapus kolom suatu matrix.
isSPLInverseable	Mengecek apakah matriks dapat diinvers.
getInverseADJ	Melakukan operasi inverse menggunakan adjoin.
perkalianWithSkalar	Melakukan perkalian scalar dengan suatu konstanta.
printInverseSPLSol	Mencetak solusi inverse sistem persamaan linear kedalam matrix.
formMatrixfrom1Col	Membuat matriks baru dari hanya 1 kolom matriks lama
getSPLGaussJordan	Mencetak hasil sistem persamaan linear dengan metode gauss jordan kedalam string.
interpolasiToString	Memberikan hasil interpolasi kedalam string.
interpolasiPolinomial	Melakukan interpolasi polinomial.
formReglin	Membentuk persamaan regresi linear berganda dari data yang diambil
chooseNGetMatrix	Prosedur untuk memilih cara menginput matrix.
chooseWriteMatrix	Opsi untuk mencetak matriks.
determinantToString	Mengubah format determian menjadi string.
chooseWriteInterpolasi	Opsi untuk mencetak matriks dalam metode interpolasi.
isToFile	Memilih metode output.
regMeasure	Menaksir nilai d
regToString	Menuliskan persamaan regresi dalam bentuk string
wantMeasure	Menanyakan apakah ingin dilakukan penaksiran dari persamaan hasil regresi

B. Image_Upscale.java

Merupakan file yang berisi method untuk upscale gambar.

Methods	Deskripsi
ImageUpscale	Prosedur utama program perbesaran gambar, menerima input dari user
imageToMatrix	Transform gambar ke matrix 3 dimensi
matrixToImage	Transform matrix menjadi image
RgbToGrayscale	Mengubah Matriks gambar RGB menjadi grayscale
getWidth	Mendapatkan jumlah pixel lebar gambar
getHeight	Mendapatkan jumlah pixel panjang gambar
fx	Mendapatkan turunan pertama terhadap x
fy	Mendapatkan turunan pertama terhadap y
fxy	Mendapatkan turunan kedua terhadap x dan y
multiplyMatrix	Mengalikan dua matriks
transposeMatrix	Mengembalikan hasil transpose matrix
getIndex	Mendapat indeks untuk pemrosesan matriks dengan batas bawah 0, batas atas 255 (range warna pixel)
copyMat	Mengembalikan copy matriks
upscale	Mengembalikan matriks dengan ukuran dikali skala perbesaran, elemen-elemen yang awalnya kosong diisi dengan algoritma bicubic spline
matrixToTxt	Output matrix menjadi file txt (untuk testing)
CheckImageExist	Mengecek apakah image ada didalam folder
checkFolderExist	Mengecek apakah folder exist
InputPathImg	Mengambil input path image dari user
OutputPathimage	Mengambil output path image dari user
inputImageName	Mengambil image name dari user
OutputImageName	Mengambil nama image untuk output dari user

C. Main.java

File utama dari source file ini. Digunakan untuk interaksi antarmuka.

Method	Deskripsi
main	Program utama dari projek ini.
spl	Mengoperasikan Sistem Persamaan Linear.
determinant	Mengoperasikan Determinan.
matBal	Mengoperasikan Invers Matriks.

intPol	Mengoperasikan Interpolasi Polinomial.
intBic	Mengoperasikan Interpolasi *Bicubic Spline*.
regLin	Mengoperasikan regresiLinear.

BAB IV

EKSPERIMEN

A. Studi Kasus Sistem Persamaan Linear

```
SPL Tidak ada solusi.
```

Gambar 6 Kasus 1a.

```
x1 = 1.00v + 3.00
x2 = 2.00v
x3 = t
x4 = 1.00v + -1.00
x5 = v
```

Gambar 7 Kasus 1b.

```
x1 = r
x2 = -1.0w + 1.0
x3 = t
x4 = -1.0w + -2.0
x5 = 1.0w + 1.0
x6 = w
```

Gambar 8 Kasus 1c.

```
x1 = 31.65830096100683
x2 = -462.95896943940215
x3 = 2101.7246430571786
x4 = -4111.774564880152
x5 = 3641.3297252962907
x6 = -1200.3899643053658

MENU
```

Gambar 9 Kasus 1d dengan n = 6.

```
x1 = 35.737665787694255
x2 = -526.5708850183953
x3 = 2069.789456337946
x4 = -2617.7045468768665
x5 = 287.08070766956826
x6 = 549.5091688759362
x7 = 369.86340771673235
x8 = -45.44167668916373
x9 = 950.7439931596857
x10 = -1087.4748267111306
```

Gambar 10 Kasus 1d dengan n = 10.

```
x1 = 1.00u + -1.00
x2 = 2.00t
x3 = t
x4 = u
```

Gambar 11 Kasus 2a.

```
x1 = 0.0
x2 = 2.0
x3 = 1.0
x4 = 1.0
```

Gambar 12 Kasus 2b.

```
x1 = -0.2243243243243243
x2 = 0.18243243243243246
x3 = 0.7094594594594594
x4 = -0.25810810810810797
```

Gambar 13 Kasus 3a.

```
SPL Tidak ada solusi.
```

Gambar 14 Kasus 3b.

```
x1 = 14.444444444444446
x2 = 7.222222222222223
x3 = 10.0
```

Gambar 15 Kasus 4.

Pada kasus 1, program menerima input sesuai dengan masukan yang tertera dalam spesifikasi tugas besar. Matriks diberikan kedalam file .txt yang dibaca oleh program. Gambar tertera merupakan hasil luaran dari input yang telah diberikan.

Pada kasus 2, program menerimpa input sesuai dengan masukan yang tertera dalam spesifikasi tugas besar. Matriks augmented dibaca dalam bentuk .txt

Pada kasus 3, persamaan diubah kedalam matriks oleh user sendiri secara manual. Matriks yang diperoleh dibuat ke dalam file .txt. Selanjutnya program mengolah data tersebut sehingga mengeluarkan data yang tertera.

Pada kasus 4, Kasus ini melibatkan isu hukum kekekalan massa di dalam sebuah reaktor, di mana aliran massa yang masuk harus setara dengan aliran massa yang keluar (mempertahankan kekekalan massa). Aliran massa diukur dengan laju massa, yang merupakan jumlah massa yang keluar atau masuk dalam satu satuan waktu. Laju massa dapat dihitung sebagai perkalian dari debit (Q = m3/s) dan massa jenis (ρ = kg/m3). Soal ini memberikan nilai Qmasuk dan Qkeluar dari masing-masing reaktor, serta tujuan untuk menemukan massa jenis zat yang masuk atau keluar. Beberapa konstanta, seperti mAin dan mCin, yang merupakan laju massa yang masuk ke reaktor A dan C, juga telah diketahui.

Terdapat tiga reaktor, dan dari data ini, terbentuk tiga persamaan linier yang mewakili laju massa pada reaktor A, B, dan C. Matriks persamaan ini telah disederhanakan dan diberikan sebagai berikut:

-120 60 0 -1300

40 -80 0 0

80 20 -150 -200

Dengan menggunakan program SPL, solusi dari matriks ini ditemukan sebagai X1 = 14.44, X2 = 7.22, dan X3 = 10. Satuan dari variabel ini tergantung pada satuan konstanta mAin dan mCin, yang dinyatakan dalam satuan mg/s. Jika dibagi dengan debit m3/s, variabel ini memiliki satuan mg/m3.

Secara formal, solusi dari studi kasus ini dapat dirumuskan sebagai berikut:

Massa jenis zat yang keluar dari reaktor A adalah 14.44 mg/m3.

Massa jenis zat yang keluar dari reaktor B adalah 7.22 mg/m3.

Massa jenis zat yang keluar dari reaktor C adalah 10 mg/m3.

Hasil ini dapat digunakan untuk mendapatkan informasi lebih lanjut tentang zat-zat dalam sistem reaktor, seperti molaritas atau konsentrasi, dengan asumsi diketahui massa molar dan sebagainya.

B. Studi Kasus Interpolasi

```
f(x) = (0.02)x^0 + (-0.57)x^1 + (5.07)x^2 + (-13.27)x^3 + (17.99)x^4 + (-11.72)x^5 + (2.93)x^6
f(0.2) = 0.028610351562498405
```

Gambar 16 Kasus 5a (1).

```
f(x) = (0.02)x^0 + (-0.57)x^1 + (5.07)x^2 + (-13.27)x^3 + (17.99)x^4 + (-11.72)x^5 + (2.93)x^6
f(0.55) = 0.17009898376464805
```

Gambar 17 Kasus 5a (2).

```
f(x) = (0.02)x^0 + (-0.57)x^1 + (5.07)x^2 + (-13.27)x^3 + (17.99)x^4 + (-11.72)x^5 + (2.93)x^6
f(0.85) = 0.344373519897462
```

Gambar 18 Kasus 5a (3).

```
f(x) = (0.02)x^0 + (-0.57)x^1 + (5.07)x^2 + (-13.27)x^3 + (17.99)x^4 + (-11.72)x^5 + (2.93)x^6
f(1.28) = 0.6720241833124909
```

Gambar 19 Kasus 5a (4).

```
dataoutput > 5b1.txt
f(x) = (9812992158028.03)x^0 + (-12343756260359.23)x^1 + (6852937863546.66)x^2 + (-2205420462593.54)x^3 + (453666728582.64)x
f(9.1666) = -681433.71875
```

Gambar 20 Kasus 5b (1).

```
dataoutput > 5b2.txt
f(x) = (9812992158028.03)x^0 + (-12343756260359.23)x^1 + (6852937863546.66)x^2 + (-2205420462593.54)x^3 + (453666728582.64)x
f(8.32258) = 36383.03515625
```

Gambar 21 Kasus 5b (2).

```
dataoutput > 5b3.txt
f(x) = (9812992158028.03)x^0 + (-12343756260359.23)x^1 + (6852937863546.66)x^2 + (-2205420462593.54)x^3 + (453666728582.64)x
f(7.51613) = 53530.361328125
```

Gambar 22 Kasus 5b (3).

```
dataoutput > 5c.txt
f(x) = (0.00)x^0 + (3.88)x^1 + (-18.81)x^2 + (57.32)x^3 + (-111.51)x^4 + (143.28)x^5 + (-123.11)x^6 + (69.96)x^7 + (-25.22)x
f(12.0) = -1.0977174800840637E10
```

Gambar 23 Kasus 5c.

Pada kasus 5a, tabel yang diberikan oleh spesifikasi disalin ke dalam bentuk matriks 8x2 dengan baris terakhir adalah angka yang ingin diaproksimasi. Program menerima input file tersebut dan melakukan operasi interpolasi polinom.

Pada kasus 5b, tanggal yang disediakan oleh spesifikasi secara manual dikalkulasikan oleh pengguna lalu ditulis secara manual ke dalam file .txt. Program menerima input file tersebut dan melakukan operasi interpolasi polinom.

Pada kasus 5c, digunakan derajat n sebesar 10. Sehingga titik yang berada pada selang tersebut di input secara manual ke dalam file .txt. Program menerima input file tersebut dan melakukan operasi interpolasi polinom.

C. Studi Kasus Regresi Linear Berganda

```
Persamaan regresi yang didapat adalah y = - 0.003x1 0.008x2 0.016x3
Nilai Aproksimasi yang didapat adalah 108
```

Gambar 24 Kasus 6

Sistem persamaan yang telah diberikan spek dimasukkan ke dalam file berbentuk (.txt). Program melakukan regresi terhadap input tersebut.

D. Studi Kasus Interpolasi *Bicubic Spline*

```
f(0.000000,0.000000) = 21.000
```

Gambar 25 Kasus 7a.

```
f(0.100000,0.900000) = 128.575
```

Gambar 26 Kasus 7b.

```
f(0.500000,0.500000) = 87.797
```

Gambar 27 Kasus 7c.

```
f(0.250000,0.750000) = 117.732
```

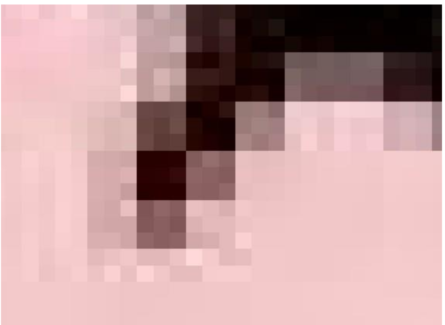
Gambar 28 Kasus 7d.

Kasus untuk bicubic spline dilakukan dengan memasukkan matriks yang telah diberikan dari spesifikasi ke dalam file (.txt). File dilakukan operasi bicubic spline serhingga menghasilkan output seperti diatas.

E. Perbesaran Gambar

Pada gambar kiri (original), gambar memiliki pixel 512 x 447, setelah di upscale dengan algoritma bicubic spline interpolation dengan skala 3, didapat gambar kanan dengan pixel 1537 x 1342. Jika kita zoom ini pada bagian mata, dapat terlihat jumlah pixel yang awalnya 1 menjadi 9.

Gambar 29 Closeup bagian mata sebelum resize.



Gambar 30 Closeup bagian mata sesudah resize.

BAB V
KESIMPULAN, SARAN, DAN REFLEKSI

A. KESIMPULAN

A. Kami telah membuat sebuah program yang mampu menghitung solusi persamaan linear dengan metode eliminasi gauss, metode eliminasi gauss – Jordan, metode matriks balikan, dan kaidah cramer.

B. Kami telah membuat sebuah program yang mampu menentukan determinan dari matriks menggunakan metode upper triangle dan kofaktor - adjoin.

C. Kami telah membuat sebuah program yang mampu menafsirkan nilai suatu angka dari beberapa titik menggunakan interpolasi polinom.

D. Kami telah membuat sebuah program yang mampu mengaproksimasi fungsi di antara titik – titik yang diketahui menggunakan metode *Bicubic Spine Interpolation*.

E. Kami telah membuat sebuah program yang mampu memprediksi nilai menggunakan regresi linear.

B. SARAN

Berdasarkan hasil akhir dari tugas besar ini, program yang dihasilkan oleh kami masih memiliki banyak kekurangan. Kekurangan tersebut antara lain penataan kode yang masih kurang efektif dan user interface yang hanya sebatas dari command line. Apabila tugas besar ini diberikan waktu yang lebih renggang, proyek ini dapat menyelesaikan tugasnya yang tertunda yaitu merapikan kode dan pembuatan GUI yang apik. Hal ini dapat membuat pengguna dapat mengakses program ini dengan mudah dan mempermudah *developer* untuk membaca kode dari proyek ini.

C. REFLEKSI

Dalam mengerjakan tugas besar ini, kami mendapatkan banyak pelajaran yang berharga. Pelajaran – pelajaran tersebut, kami yakin, dapat membantu kami untuk menempuh perkuliahan di Teknik Informatika Institut Teknologi Bandung.

Pelajaran pertama ialah kurangnya pengetahuan kami terhadap *Object Oriented Programming*. Pada saat tugas besar ini rilis, kami tidak dapat langsung menggunakan java seperti bahas pemrograman yang telah kami pelajari. Kami harus mempelajari dasar dari pemrograman berorientasi objek sehingga terdapat *learning curve* yang harus kami lalui. Waktu yang ditempuh untuk mempelajari paradigma pemrograman tersebut sangatlah banyak dan menghabiskan waktu pengerjaan kami.

Pelajaran kedua adalah ketidakefisienannya kami dalam mengatur waktu untuk mengerjakan tugas besar ini. Kami cenderung menunda-nunda pengerjaan projek ini sehingga pada saat waktu dekat pengumpulan kami harus bekerja lebih ekstra dibandingkan biasanya.

Masih ada pelajaran – pelajaran lainnya yang tidak bisa kami sebutkan satu – satu. Kesalahan yang kami lakukan murni datang dari kekurangan dan kemampuan kami. Oleh karena itu, kami akan terus berusaha hingga segala kekurangan yang kami miliki tidak akan menjadi hambatan untuk hari yang akan datang.


Programmers while reviewing the codes
Why?
Why?
Why?
Oh, that's why.

BAB VI
REFERENSI

https://www.mssc.mu.edu/~daniel/pubs/RoweTalkMSCS_BiCubic.pdf

https://stackoverflow.com/questions/3612567/how-to-create-my-own-java-libraryapi

https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/algeo23-24.htm

BAB VII
PEMBAGIAN TUGAS

No.	**Bagian Program**	**Sub-Bagian Program**	**Penanggung Jawab**	**Status**
1.	Sistem Persamaan Linear	Metode Eliminasi Gauss	Rayhan, Althaf	CLEAR(20/9)
		Metode Eliminasi Gauss - Jordan	Rayhan, Rafly	CLEAR(20/9)
		Metode Balikan	Rayhan	CLEAR(20/9)
		Metode Cramer	Rayhan, Rafly	CLEAR(21/9)
2.	Determinan	Metode Eliminasi Gauss (Upper Triangle)	Rafly	CLEAR(24/9)
		Metode Ekspansi Kofaktor	Rafly	CLEAR(24/9)
3.	Matriks Balikan	Metode Eliminasi Gauss – Jordan (Identitas)	Rafly, Althaf	CLEAR(27/9)
		Metode Ekspansi Kofaktor dan Adjoin	Althaf	CLEAR(26/9)
4.	Interpolasi Polinom	Aplikasi Interpolasi Polinom	Rafly	CLEAR(27/9)
5.	Interpolasi Bicubic Spline	Aplikasi Interpolasi Bicubic Spline	Althaf	CLEAR(6/10)
		Pemrosesan Citra Gambar	Rayhan, Althaf	CLEAR(5/10)
6.	Regresi Linear Berganda	Aplikasi Linear Berganda	Althaf	CLEAR(6/10)