

Spesifikasi Tugas Besar 1

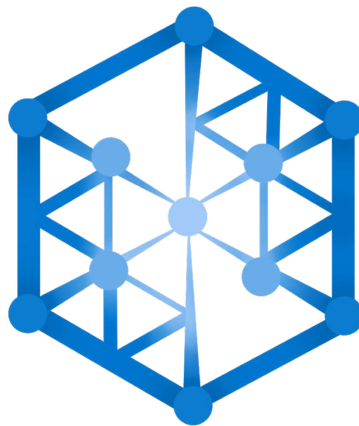
IF2123 ALJABAR LINIER DAN GEOMETRI

SISTEM PERSAMAAN LINIER, DETERMINAN, DAN PENERAPANNYA

SEMESTER I TAHUN 2026/2027

v1.0

DEADLINE: RABU, 2026/9/30 21:23 WIB



LABORATORIUM ILMU DAN REKAYASA KOMPUTASI

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

Dokumen spesifikasi tugas besar ini disusun oleh asisten Laboratorium Ilmu dan Rekayasa Komputasi dari mahasiswa Teknik Informatika angkatan 2024. Untuk berjaga-jaga dengan adanya revisi, salinan terbaru dari dokumen ini selalu dapat diakses di [sini](#).

0 Daftar Revisi

- v1.0 — [2026/09/09 13.00] — Versi rilis awal.

Daftar Isi

0	Daftar Revisi	1
1	Pendahuluan	2
2	Dasar Teori	3
2.1	Sistem Persamaan Linear	3
2.2	Determinan Matriks	3
2.3	Invers Matriks	3
2.4	Interpolasi Polinomial	3
2.5	Natural Cubic Spline Interpolation	4
2.6	Splines Regression	5
3	Spesifikasi	7
3.1	Spesifikasi Wajib Program	7
3.2	Spesifikasi Bonus	13
3.3	Prosedur Pengerjaan	14
3.4	Spesifikasi Laporan	15
3.5	Kriteria Penilaian	17
4	Pengumpulan	18
4.1	<i>Deliverables</i>	18
4.2	Teknis Pengumpulan	18
5	Studi dan Kasus Uji	20
5.1	Determinan Matriks	20
5.2	Invers Matriks	20
5.3	Sistem Persamaan Linier ($Ax = b$)	20
5.4	Sistem Persamaan Linier Berbentuk Matriks Augmented	21
5.5	Sistem Persamaan Linier Berbentuk Umum	22
5.6	Aplikasi Sistem Persamaan Linear	22
5.7	Interpolasi Polinomial	22
5.8	Studi Interpolasi Natural Cubic Spline	23
5.9	Studi Kasus Regresi Spline	23
6	Referensi	24
6.1	SPL, Determinan, dan Matriks	24
6.2	Interpolasi Polinomial	24
6.3	Natural Cubic Spline Interpolation	24
6.4	Regresi Spline (<i>Regression Splines</i>)	24
6.5	Image Inpainting dengan Persamaan Laplace	24
7	Surat Cinta dari Asisten	25

1 Pendahuluan

Di pelosok Snezhnaya, seorang peneliti muda bernama **Mitya** tengah berusaha keras memimpin eksperimen krusial berskala raksasa: **Proyek Stuzha**. Proyek ini digadang-gadang mampu mengendalikan fluktuasi energi ekstrem di daratan Teyvat. Sayangnya, impian besar tersebut terbentur tembok tebal: Mitya masih mengandalkan kalkulator saku tua yang sering *error* tiap kali dipaksa menyelesaikan kalkulasi berdimensi tinggi.



Gambar 1: Berkenalan dengan Mitya

Sumber : Arsip Penulis

Setelah menjelajahi perpustakaan Sumeru Akademiya hingga menyisir kabar burung ke seantero Teyvat, Mitya mendengar reputasi kalian sebagai praktisi komputasi dan aljabar linier paling piawai. Ia pun mengirimkan pesan ke Adventures Guild dan memberikan komisi **perangkat lunak sekelas Photomath versi Teyvat** yaitu sebuah mesin kalkulasi otomatis yang sanggup melumat sistem matriks dan persamaan linier rumit demi kelangsungan Proyek Stuzha.

Aljabar linier bukan sekadar teori di atas kertas papirus; ini adalah fondasi komputasi modern yang menggerakkan simulasi rekayasa, pemodelan aliran energi, analisis data, hingga dinamika sistem kompleks. Agar kalkulasi Proyek Stuzha tidak runtuh di tengah jalan, kalian ditugaskan merancang dan membangun pustaka (*library*) komputasi aljabar linier mandiri berbasis bahasa pemrograman **Java** yang modular, tangguh, dan terdokumentasi rapi.

Pustaka ini nantinya menjadi motor utama di balik alat komputasi Mitya untuk:

- Menyelesaikan **Sistem Persamaan Linier (SPL)**, perhitungan **Determinan**, dan pencarian **Invers Matriks** guna menganalisis keseimbangan sirkuit daya dan kestabilan instrumen Proyek Stuzha.
- Melakukan **Interpolasi Polinomial**, *Natural Cubic Spline Interpolation*, dan regresi *Spline* untuk memetakan jalur transfer energi serta memodelkan kontur data eksperimen secara presisi.
- Menerapkan *Image Inpainting (Hole Filling)* berbasis **Persamaan Laplace (Bonus)** sebagai modul restorasi darurat. Modul ini bertugas merekonstruksi lembar cetak biru dan catatan eksperimen penting yang kerap berlubang atau hilang secara misterius yang konon merupakan ulah sabotase usil dari **Renova**.

Melalui tugas besar ini, kalian diharapkan tidak hanya menguasai teori matematika di balik algoritma-algoritma tersebut, melainkan juga mampu mengonstruksinya menjadi perangkat lunak nyata yang lolos berbagai skenario uji (*test cases*) ekstrem.

Tujuan Tugas Besar:

- Mengimplementasikan metode penyelesaian SPL, determinan, invers matriks, interpolasi polinomial, *Natural Cubic Spline Interpolation*, regresi *Spline*, dan algoritma *Image Inpainting* berbasis diskritisasi Persamaan Laplace secara mandiri menggunakan Java.
- Membangun pustaka (*library*) aljabar linier yang terstruktur secara modular, bersih, dan terdokumentasi dengan baik. Tidak menggunakan *library* perhitungan matriks, regresi, atau interpolasi instan.
- Menguji keandalan pustaka pada berbagai kasus uji analitis dan restorasi citra, serta menyusun laporan analisis hasil perhitungannya.

2 Dasar Teori

2.1 Sistem Persamaan Linear

Sistem Persamaan Linear (SPL) adalah himpunan berhingga dari persamaan-persamaan linear dengan variabel-variabel yang sama. Bentuk umum dari SPL dengan m persamaan dan n variabel adalah

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n = b_n \end{cases}$$

atau, dalam bentuk perkalian matriks (kiri) dan matriks augmented (kanan)

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}}_A \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_x = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}}_b \rightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{bmatrix}$$

Solusi dari SPL dapat berupa solusi tunggal, banyak solusi, atau tidak ada solusi. Metode penyelesaian yang umum digunakan antara lain eliminasi Gauss, eliminasi Gauss-Jordan, kaidah Cramer, dan metode matriks balikan.

2.2 Determinan Matriks

Determinan adalah sebuah nilai skalar khusus yang dapat dihitung dari elemen-elemen sebuah matriks persegi. Determinan dari matriks A dinotasikan sebagai $\det(A)$ atau $|A|$. Nilai determinan sangat krusial karena dapat mengindikasikan apakah matriks tersebut memiliki invers. Beberapa metode untuk menghitung determinan matriks mencakup metode ekspansi kofaktor dan metode reduksi baris.

2.3 Invers Matriks

Invers dari sebuah matriks persegi A adalah matriks A^{-1} yang memenuhi properti $AA^{-1} = A^{-1}A = I$, di mana I adalah matriks identitas. Sebuah matriks hanya memiliki invers jika dan hanya jika determinannya tidak nol ($\det(A) \neq 0$). Matriks yang memiliki invers disebut matriks *invertible* atau non-singular. Perhitungan invers dari suatu matriks dapat dilakukan dengan metode matriks adjoint maupun metode eliminasi Gauss-Jordan.

2.4 Interpolasi Polinomial

Interpolasi polinomial adalah metode untuk menentukan suatu polinomial $P_n(x)$ berderajat n yang melalui $n+1$ titik data $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, x_i diurutkan dengan x_0 terkecil dan x_n terbesar. Secara umum, polinomial tersebut dapat dituliskan sebagai:

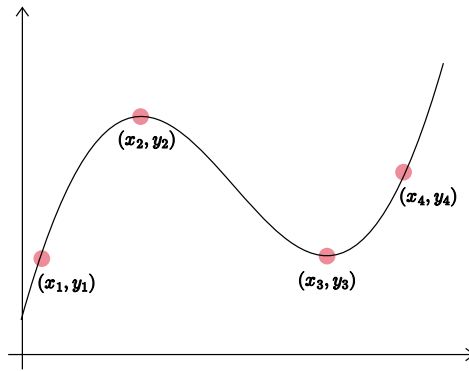
$$P_n(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$$

dengan koefisien $a_0, a_1, \dots, a_n$ yang ditentukan sedemikian rupa sehingga $P_n(x_{\text{perkiraan}}) = y_{\text{perkiraan}}$ untuk semua nilai $x_0 < x_{\text{perkiraan}} < x_n$. Derajat persamaan polinom ditentukan oleh jumlah titik data yang diberikan dikurangi satu.

Koefisien polinomial dapat dicari dengan menyulihkan titik-titik data ke dalam persamaan polinomial dan menyelesaikan sistem persamaan linear yang dihasilkan dengan metode eliminasi Gauss yang dijelaskan sebelumnya. Kita dapat menyusun sistem persamaan linear sebagai berikut.

$$\begin{cases} a_0 + a_1x_0 + a_2x_0^2 + \cdots + a_nx_0^n = y_0 \\ a_0 + a_1x_1 + a_2x_1^2 + \cdots + a_nx_1^n = y_1 \\ \vdots \\ a_0 + a_1x_n + a_2x_n^2 + \cdots + a_nx_n^n = y_n \end{cases}$$

Persamaan-persamaan di atas dapat dipetakan menjadi sebuah permasalahan sistem persamaan linear, sehingga dapat diselesaikan dengan menggunakan metode eliminasi Gauss untuk menyelesaikan

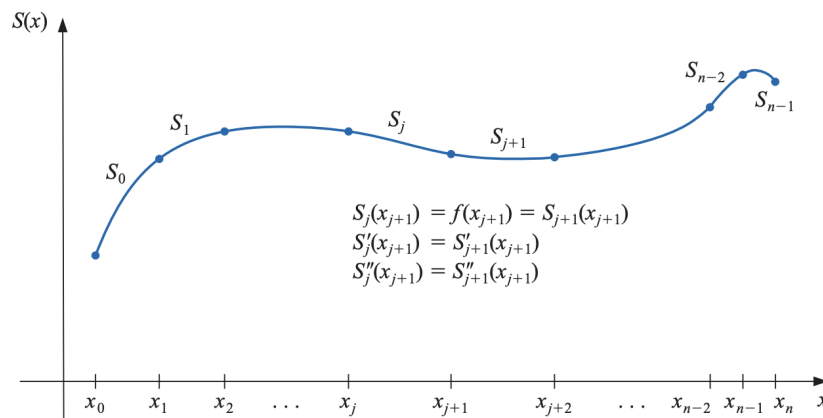


Gambar 2: Contoh interpolasi polinomial melalui empat titik data.

sistem tersebut dan menemukan koefisien polinomial $a_0, a_1, \dots, a_n$. Dari polinom $P_n(x)$ yang dibentuk dengan koefisien tersebut, kita dapat memperkirakan nilai y untuk nilai x yang berada di antara titik-titik data yang diberikan (interpolasi) maupun di luar titik-titik data tersebut (ekstrapolasi).

2.5 Natural Cubic Spline Interpolation

Interpolasi Spline Kubik (Cubic Spline Interpolation) adalah metode interpolasi yang mengonstruksi kurva mulus melalui sekumpulan titik data diskret menggunakan potongan-potongan polinomial berderajat tiga (kubik) di setiap interval antar-titik. Berbeda dengan interpolasi polinomial tunggal orde tinggi yang rentan mengalami osilasi liar di ujung interval, *spline* menjaga kelengkungan kurva tetap stabil dan kontinu hingga turunan kedua.



Gambar 3: Contoh Interpolasi Spline Kubik
Sumber: R. L. Burden & J. D. Faires, *Numerical Analysis*

Definisi dari *cubic spline interpolation* S untuk sebuah fungsi f yang terdefinisi di $[a, b]$ dengan sekumpulan titik data **terurut** $a = x_0 < x_1 < \dots < x_n = b$ memenuhi kondisi berikut:

- $S(x)$ merupakan potongan polinomial kubik $S_j(x)$ pada setiap subinterval $[x_j, x_{j+1}]$ untuk $j = 0, 1, \dots, n-1$ (terdapat n subinterval).
- $S_j(x_j) = f(x_j)$ dan $S_j(x_{j+1}) = f(x_{j+1})$ untuk setiap $j = 0, 1, \dots, n-1$.
(Kurva harus melalui nilai fungsi pada titik-titik ujung tiap subinterval).
- $S_{j+1}(x_{j+1}) = S_j(x_{j+1})$ untuk setiap $j = 0, 1, \dots, n-2$.
(Kurva bersambung secara kontinu tanpa putus di setiap simpul interior).
- $S'_{j+1}(x_{j+1}) = S'_j(x_{j+1})$ untuk setiap $j = 0, 1, \dots, n-2$.
(Kemiringan kurva dari sisi kiri dan kanan bernilai sama di setiap simpul interior).
- $S''_{j+1}(x_{j+1}) = S''_j(x_{j+1})$ untuk setiap $j = 0, 1, \dots, n-2$.
(Kelengkungan (*curvature*) kurva dari sisi kiri dan kanan bernilai sama di setiap simpul interior).

f. $S''(x_0) = S''(x_n) = 0$ untuk *natural spline*.

(Kurva tidak memiliki gaya lentur pada kedua ujung terluar sehingga berperilaku menyerupai garis lurus di luar rentang data).

Penjelasan matematis lebih mendalam mengenai penurunan rumus, pembentukan sistem persamaan tridiagonal, serta visualisasi grafik interpolasi spline kubik dapat dirujuk pada buku *Numerical Analysis* karya Burden dan Faires.

2.6 Splines Regression

Splines Regression adalah metode pemodelan non-parametrik yang memetakan hubungan non-linear antara variabel prediktor x dan variabel respons y menggunakan kombinasi potongan polinomial terhubung pada simpul-simpul tertentu (*knots*), disimbolkan dengan $\xi_1 < \xi_2 < \dots < \xi_K$. Berbeda dari interpolasi yang menyelesaikan SPL deterministik untuk menghubungkan turunan pada setiap titik data, pendekatan *splines regression* mentranslasikan permasalahan ke dalam Sistem Persamaan Linear *Overdetermined* (jumlah titik data $N >$ jumlah basis M) melalui kerangka ruang fungsi basis, kemudian diselesaikan menggunakan Metode Kuadrat Terkecil (*Ordinary Least Squares*).

1. Konstruksi Ruang Fungsi dan Basis Daya Terpancung (*Truncated Power Basis*)

Ruang fungsi spline kubik dengan K simpul internal $\xi_1, \xi_2, \dots, \xi_K$ merupakan ruang vektor berdimensi $M = 4 + K$, di mana 4 adalah jumlah koefisien polinomial kubik standar dan K adalah jumlah fungsi basis tambahan untuk setiap simpul.

Setiap fungsi $f(x)$ dalam ruang tersebut dapat dinyatakan sebagai kombinasi linear unik dari M buah fungsi basis $\{h_0(x), h_1(x), h_2(x), h_3(x), \dots, h_{3+K}(x)\}$:

Polinomial standar global:

$$h_0(x) = 1, \quad h_1(x) = x, \quad h_2(x) = x^2, \quad h_3(x) = x^3$$

Basis tambahan di setiap simpul ξ_k :

$$h_{3+k}(x) = (x - \xi_k)_+^3 = \begin{cases} (x - \xi_k)^3, & \text{jika } x > \xi_k \\ 0, & \text{jika } x \leq \xi_k \end{cases} \quad \text{untuk } k = 1, 2, \dots, K$$

Fungsi regresi menjadi:

$$f(x) = \sum_{j=0}^{3+K} \beta_j h_j(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \sum_{k=1}^K \beta_{3+k} (x - \xi_k)_+^3$$

Sifat kontinuitas turunan hingga C^2 di seluruh rentang x langsung terpenuhi secara analitis oleh bentuk fungsi basis $(x - \xi_k)_+^3$ tanpa memerlukan penambahan persamaan batas turunan manual.

Catatan notasi:

$$(u)_+ = \max(0, u) = \begin{cases} u, & \text{jika } u > 0 \\ 0, & \text{jika } u \leq 0 \end{cases}$$

2. Formulasi Matriks Desain ($X\beta \approx y$)

Diberikan N sampel data pengamatan $(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)$ dengan kondisi jumlah data jauh lebih banyak daripada jumlah basis ($N > M$). Persamaan untuk setiap titik observasi $i = 1, 2, \dots, N$ dituliskan sebagai:

$$y_i = \beta_0 h_0(x_i) + \beta_1 h_1(x_i) + \dots + \beta_{3+K} h_{3+K}(x_i) + \varepsilon_i$$

Dalam notasi matriks-vektor aljabar linear:

$$y = X\beta + \varepsilon$$

dengan:

- Vektor observasi: $y = [y_1 \ y_2 \ \dots \ y_N]^T \in \mathbb{R}^{N \times 1}$
- Vektor parameter bobot: $\beta = [\beta_0 \ \beta_1 \ \dots \ \beta_{3+K}]^T \in \mathbb{R}^{M \times 1}$
- Matriks Desain $X \in \mathbb{R}^{N \times M}$:

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & (x_1 - \xi_1)_+^3 & (x_1 - \xi_2)_+^3 & \dots & (x_1 - \xi_K)_+^3 \\ 1 & x_2 & x_2^2 & x_2^3 & (x_2 - \xi_1)_+^3 & (x_2 - \xi_2)_+^3 & \dots & (x_2 - \xi_K)_+^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & x_N^2 & x_N^3 & (x_N - \xi_1)_+^3 & (x_N - \xi_2)_+^3 & \dots & (x_N - \xi_K)_+^3 \end{bmatrix}$$

3. Proyeksi Ortogonal dan Persamaan Normal

Karena sistem $X\beta = \mathbf{y}$ bersifat *overdetermined* (baris $N >$ kolom M), vektor respons $\mathbf{y}$ umumnya tidak terletak di dalam ruang kolom matriks X ($\text{Col}(X)$). Solusi terbaik diperoleh dengan mencari vektor taksiran $\hat{\mathbf{y}} = X\hat{\beta}$ yang merupakan proyeksi ortogonal dari $\mathbf{y}$ ke subruang $\text{Col}(X)$, meminimalkan norma kuadrat vektor residu $\|\mathbf{e}\|^2$:

$$\min_{\beta} \|\mathbf{y} - X\beta\|^2 = (\mathbf{y} - X\beta)^T (\mathbf{y} - X\beta)$$

Berdasarkan teorema proyeksi aljabar linear, vektor sisaan $\mathbf{e} = \mathbf{y} - X\hat{\beta}$ harus tegak lurus (ortogonal) terhadap seluruh kolom X :

$$X^T (\mathbf{y} - X\hat{\beta}) = \mathbf{0}$$

Persamaan ini menghasilkan Persamaan Normal (*Normal Equations*):

$$(X^T X)\hat{\beta} = X^T \mathbf{y}$$

Estimasi Parameter Koefisien

Jika kolom-kolom pada matriks X bebas linear (*linearly independent*), matriks Gram $X^T X$ berukuran $M \times M$ bersifat simetris definit positif dan memiliki invers, sehingga solusi parameter koefisien $\hat{\beta}$ ditentukan secara unik:

$$\hat{\beta} = (X^T X)^{-1} X^T \mathbf{y}$$

Penjelasan lebih mendalam mengenai formulasi matematis, fungsi basis terpancung, serta konsep regularisasi regresi spline dapat dirujuk pada buku *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* karya Hastie, Tibshirani, dan Friedman.

3 Spesifikasi

3.1 Spesifikasi Wajib Program

Secara ringkas, luaran minimum yang diharapkan pada tugas besar ini terdapat tiga:

1. Buatlah pustaka (*library* atau *package*) dalam bahasa Java untuk segala perhitungan matriks, regresi, dan interpolasi yang disebut di bagian 2. Bentuk dari pustaka ini adalah file `.jar` yang dapat digunakan pada program Java lainnya.
2. Gunakan pustaka yang dibuat pada poin 1 untuk membuat program berbasis **antarmuka baris perintah** (*command line interface*, CLI) atau **antarmuka pengguna grafis** (*graphical user interface*, GUI) yang memperlihatkan fungsionalitas dari pustaka tersebut disertai dengan dokumentasi **README** pada repository GitHub. Untuk setiap fitur wajib, program dapat menerima masukan secara manual (input keyboard) maupun dari berkas `.txt` dengan format yang ditentukan.
3. Laporan mengenai pustaka dan program yang dibuat.

3.1.1 Batasan Umum Tugas Besar

1. Program wajib dikembangkan menggunakan bahasa pemrograman **Java** dengan versi minimal Java 8 (diperbolehkan menggunakan versi yang lebih baru seperti Java 11, 17, 21, atau versi rilis terkini).
2. Seluruh program dan pustaka wajib dikelola serta dibangun menggunakan **Apache Maven**, dan harus dapat dikemas menjadi berkas `.jar` yang siap dieksekusi.
3. Seluruh pustaka aljabar linier wajib dibangun mandiri tanpa pustaka eksternal (seperti JAMA, EJML, atau Commons Math). Penggunaan `java.lang.Math` hanya diizinkan untuk fungsi dasar (`sqrt`, `exp`, `sin`, `cos`).
4. Struktur data matriks wajib dirancang dan diimplementasikan sendiri dari awal (misalnya kelas **Matrix** dengan atribut array `double[] [] data`), termasuk operasi-operasi dasarnya seperti penjumlahan, pengurangan, perkalian, transpose, serta pembentukan submatriks.
5. Program harus memiliki mekanisme penanganan kesalahan (*error handling*) yang tangguh agar tidak mengalami *crash* akibat galat masukan maupun komputasi. Penanganan kesalahan wajib memanfaatkan mekanisme **Java Exception** (blok `try-catch`) untuk mengatasi:
 - a. Masukan pengguna yang tidak valid (misalnya memasukkan karakter huruf saat program meminta angka).
 - b. Dimensi matriks yang tidak konsisten atau tidak sesuai (misalnya jumlah kolom antarbeberapa baris berbeda).
 - c. Berkas masukan yang tidak ditemukan atau tidak dapat dibuka (`FileNotFoundException`).
 - d. Ketidaksesuaian format isi berkas (misalnya adanya baris kosong atau susunan elemen yang keliru).
 - e. Operasi matematika yang tidak terdefinisi (misalnya invers pada matriks singular atau perhitungan determinan matriks non-persegi).
6. Program harus mampu menerima tanda desimal berupa titik (.) maupun koma (,) pada data masukan. Untuk tampilan keluaran, seluruh nilai desimal dibulatkan hingga maksimal 3 angka di belakang koma. Format masukan diasumsikan tidak memuat pemisah ribuan.
7. Batasan ukuran masukan yang harus didukung oleh program meliputi:
 - a. Untuk masukan manual dari *keyboard*, dimensi matriks dibatasi maksimal 11×11 (kolom konstanta pada SPL tidak dihitung sebagai bagian dari dimensi matriks utama, sehingga matriks *augmented* maksimal berukuran 11×12).
 - b. Untuk masukan melalui berkas `.txt`, program dapat memproses matriks hingga ukuran maksimal 1001×1001 (atau 1001×1002 untuk matriks *augmented* SPL).
 - c. Untuk modul interpolasi dan regresi *spline*, jumlah titik data sampel yang diberikan dibatasi maksimal 10 titik.

3.1.2 Rincian Pustaka (*Library*)

Pustaka aljabar linier yang dibuat minimal harus menyediakan modul atau fungsi sebagai berikut:

1. Sistem Persamaan Linier (SPL)

- Eliminasi Gauss (menghasilkan matriks eselon baris, dilanjutkan dengan substitusi mundur).
- Eliminasi Gauss-Jordan (menghasilkan matriks eselon baris tereduksi).
- Metode Matriks Balikan ($x = A^{-1}b$).
- Kaidah Cramer (khusus untuk SPL dengan jumlah persamaan = jumlah variabel dan $\det(A) \neq 0$).
- Untuk menjaga stabilitas numerik**, metode eliminasi Gauss dan Gauss-Jordan **WAJIB** menerapkan **Partial Pivoting** (pemilihan elemen pivot terbesar dari kolom yang sedang diproses) guna meminimalkan galat pembulatan, terutama pada matriks dengan kondisi buruk (*ill-conditioned*), seperti matriks Hilbert.

2. Determinan Matriks

- Metode Reduksi Baris / OBE (dengan memperhatikan efek pertukaran baris dan perkalian baris terhadap nilai determinan).
- Metode Ekspansi Kofaktor (dapat memilih baris atau kolom acuan untuk efisiensi).

3. Matriks Balikan (Invers)

- Metode Augmentasi $[A \mid I]$ dengan Eliminasi Gauss-Jordan.
- Metode Adjoin (menggunakan Matriks Kofaktor dan determinan).

4. Interpolasi Polinomial

Menghitung koefisien polinomial interpolasi berderajat $n - 1$ yang melalui seluruh pasangan titik data masukan (x_i, y_i) .

5. Natural Cubic Spline Interpolation (Fitur Wajib)

- Membangun polinomial kubik *piecewise* $S_j(x)$ untuk setiap interval $[x_j, x_{j+1}]$ yang melalui semua titik data.
- Menyelesaikan sistem persamaan linear tridiagonal yang terbentuk untuk mencari turunan kedua M_i pada setiap titik.
- Menerapkan syarat batas natural ($M_0 = M_{n-1} = 0$).

6. Regresi Spline Kubik (*Cubic Regression Splines*) (Fitur Wajib)

- Membangun basis spline kubik dengan *truncated power basis* berdasarkan posisi *knot* yang ditentukan pengguna.
- Menyusun matriks desain X dan menyelesaikan Persamaan Normal $(X^T X)\beta = X^T y$ untuk mendapatkan koefisien regresi.

3.1.3 Rincian Umum Program

Program utama wajib memiliki antarmuka *Command Line Interface* (CLI) dengan sistem menu. Program dapat menerima input dari keyboard (manual) maupun dari file **.txt**. Semua output harus dapat ditampilkan di layar dan disimpan ke dalam file **.txt** sesuai permintaan pengguna.

Struktur Menu Minimum (dapat dikembangkan lebih lanjut):

MENU UTAMA

- Sistem Persamaan Linier (SPL)
- Determinan Matriks
- Matriks Balikan (Invers)
- Interpolasi Polinomial
- Natural Cubic Spline Interpolation
- Regresi Spline Kubik
- Keluar

Untuk menu nomor 1, 2, 3, 4, dan 6, wajib terdapat sub-menu untuk memilih metode yang tersedia sesuai dengan daftar pada Bagian 3.1.2. Contoh sub-menu untuk menu nomor 1:

SUB-MENU SPL

1. Metode Eliminasi Gauss
2. Metode Eliminasi Gauss-Jordan
3. Metode Matriks Balikan
4. Kaidah Cramer

Program wajib menampilkan langkah-langkah perhitungan (tahapan OBE, matriks yang terbentuk, dan lain-lain) secara informatif agar pengguna dapat memahami alur komputasi yang dilakukan.

3.1.4 Rincian Input dan Output Program

Untuk setiap modul, masukan dapat diberikan secara manual atau melalui file `.txt` dengan format yang dijelaskan pada bagian berikut. Jika format isi file tidak sesuai, wajib dilakukan *error handling*. Hasil perhitungan ditampilkan pada program dan dapat disimpan ke dalam file `.txt`. Nama file tidak boleh digunakan untuk menentukan maupun memvalidasi jenis masukan.

A. Sistem Persamaan Linier (SPL)

Input:

- Matriks augmented yang dimasukkan secara manual atau melalui file `.txt` dengan format tertera di bawah (setiap bilangan dipisahkan oleh *whitespace*).
- Metode penyelesaian SPL sesuai daftar pada Bagian 3.1.2.

Format Input File SPL:

Nama file: `spl_xxx.txt`

Keterangan: `xxx` merupakan deskripsi bebas dari isi file.

Format isi file:

<code>a11</code>	<code>a12</code>	<code>...</code>	<code>a1n</code>	<code>b1</code>
<code>a21</code>	<code>a22</code>	<code>...</code>	<code>a2n</code>	<code>b2</code>
<code>...</code>	<code>...</code>	<code>...</code>	<code>...</code>	<code>...</code>
<code>am1</code>	<code>am2</code>	<code>...</code>	<code>amn</code>	<code>bm</code>

Contoh:

<code>3</code>	<code>4.5</code>	<code>2.8</code>	<code>10</code>	<code>12</code>
<code>-3</code>	<code>7</code>	<code>8.3</code>	<code>11</code>	<code>-4</code>
<code>0.5</code>	<code>-10</code>	<code>-9</code>	<code>12</code>	<code>0</code>

Output:

- Langkah-langkah penyelesaian sesuai metode.
- Solusi SPL:
 - Jika solusi tunggal: tampilkan nilainya.
 - Jika tidak ada solusi: tampilkan *“solusi tidak ada”*.
 - Jika terdapat tak hingga solusi: tampilkan dalam bentuk parametrik dengan pemilihan simbol untuk variabel parametrik dibebaskan. Contoh:

$$x_1 = t, \quad x_2 = s, \quad x_3 = s - t, \quad x_4 = s + t$$

Untuk menangani kasus jika variabel parametrik terlalu banyak sehingga kita tidak punya cukup huruf untuk mencakup semuanya, boleh menggunakan notasi a_i (atau pilih sembarang huruf lainnya) sebagai variabel parametrik. Contoh: $x_2 = 2a_3 + a_5$.

- Output dapat disimpan dalam file `.txt`, memuat:
 - Metode penyelesaian SPL
 - Input matriks yang digunakan
 - Hasil SPL

B. Determinan

Input:

- Matriks yang dimasukkan secara manual atau melalui file `.txt` dengan format tertera di bawah (setiap bilangan dipisahkan oleh *whitespace*).
- Metode perhitungan determinan sesuai daftar pada Bagian 3.1.2.

Format Input File Determinan:

Nama file: `det_xxx.txt`

Keterangan: `xxx` merupakan deskripsi bebas dari isi file.

Format isi file:

```
a11  a12  ...  a1n
a21  a22  ...  a2n
...  ...  ...  ...
an1  an2  ...  ann
```

Contoh:

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

Output:

- Langkah-langkah perhitungan sesuai metode.
- Nilai determinan matriks, keluarkan *“matriks tidak memiliki determinan”* jika memang matriks input tidak memiliki determinan.
- Output dapat disimpan dalam file `.txt`, memuat:
 - Metode perhitungan determinan
 - Input matriks yang digunakan
 - Hasil determinan

C. Matriks Balikan

Input:

- Matriks yang dimasukkan secara manual atau melalui file `.txt` dengan format tertera di bawah (setiap bilangan dipisahkan oleh *whitespace*).
- Metode pencarian matriks balikan sesuai daftar pada Bagian 3.1.2.

Format Input File Matriks Balikan:

Nama file: `inverse_xxx.txt`

Keterangan: `xxx` merupakan deskripsi bebas dari isi file.

Format isi file:

```
a11  a12  ...  a1n
a21  a22  ...  a2n
...  ...  ...  ...
an1  an2  ...  ann
```

Contoh:

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

Output:

- Langkah-langkah perhitungan sesuai metode.
- Matriks balikan, keluarkan “*matriks tidak memiliki balikan*” jika memang matriks input tidak memiliki balikan.
- Output dapat disimpan dalam file `.txt`, memuat:
 - Metode perhitungan matriks balikan
 - Input matriks yang digunakan
 - Hasil matriks balikan

D. Interpolasi

Input:

- Jumlah titik sampel n . Untuk masukan melalui file, nilai n ditentukan berdasarkan jumlah baris data.
- Pasangan titik sampel $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ yang dimasukkan secara manual atau file `.txt` dengan format tertera di bawah.
- Metode interpolasi sesuai daftar pada Bagian 3.1.2.

Format Input File Interpolasi:

Nama file: `interpolasi_xxx.txt`

Keterangan: `xxx` merupakan deskripsi bebas dari isi file.

Format isi file:

```
x1  y1
x2  y2
...  ...
xn  yn
```

Contoh:

```
0    1
1    2
2    0
3    1
```

Output (interpolasi polinomial):

- Koefisien dan domain interpolasi.

- Persamaan interpolasi $P(x)$.
- Dari persamaan yang dibentuk, program dapat menerima input nilai-nilai x_t dalam domain interpolasi untuk mengevaluasi nilai $P(x_t)$.
- Output dapat disimpan dalam file `.txt`, memuat:
 - Metode interpolasi
 - Input titik sampel yang digunakan
 - Domain interpolasi
 - Persamaan hasil interpolasi $P(x)$
 - Hasil evaluasi $P(x_t)$

Output (interpolasi splina kubik natural):

- Nilai turunan kedua pada setiap knot.
- Persamaan polinom kubik untuk setiap segmen $S_i(x)$.
- Domain setiap segmen.
- Dari persamaan yang dibentuk, program dapat menerima input nilai-nilai x_t dalam domain interpolasi untuk mengevaluasi nilai $S(x_t)$.
- Output dapat disimpan dalam file `.txt`, memuat:
 - Metode interpolasi
 - Input titik sampel yang digunakan
 - Domain interpolasi
 - Persamaan hasil interpolasi setiap segmen $S_i(x)$
 - Hasil evaluasi $S(x_t)$

E. Regresi Splina Kubik

Input:

- Jumlah titik sampel n . Untuk masukan melalui file, nilai n ditentukan berdasarkan jumlah baris data.
- Pasangan titik sampel $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ yang dimasukkan secara manual atau file `.txt` dengan format tertera di bawah.
- Setelah sampel data diterima, program dapat menerima input berikut:
 - Jumlah knot K .
 - Posisi knot $k_1, k_2, \dots, k_K$.

Format Input File Regresi:

Nama file: `regresi_xxx.txt`

Keterangan: xxx merupakan deskripsi bebas dari isi file.

Format isi file:

```
x1  y1
x2  y2
...  ...
xn  yn
```

Contoh:

```
1    2.3
2    3.1
3    4.8
4    5.2
5    6.1
6    7.0
```

Output:

- Koefisien regresi.
- Posisi knot yang digunakan.
- Persamaan regresi splina $\hat{y}(x)$.
- Dari persamaan yang dibentuk, program dapat menerima input nilai-nilai x_t untuk memprediksi nilai $\hat{y}(x_t)$.
- Output dapat disimpan dalam file `.txt`, memuat:
 - Input titik sampel yang digunakan
 - Posisi knot
 - Persamaan hasil regresi $\hat{y}(x)$
 - Hasil prediksi $\hat{y}(x_t)$

3.2 Spesifikasi Bonus

Mahasiswa berkesempatan untuk memperoleh nilai tambahan (dengan batas akumulasi nilai bonus maksimal 20 poin) dengan mengerjakan fitur-fitur di bawah ini. Bonus diklaim melalui **demo langsung** kepada asisten serta pemeriksaan kode sumber. Seluruh fitur bonus yang dikembangkan wajib terintegrasi dengan pustaka dan program utama yang telah dibuat.

3.2.1 Antarmuka Pengguna Grafis (APG/GUI)

Silakan gunakan GUI jika ingin membuat aplikasi yang lebih interaktif. Perlu diingat bahwa tugas besar ini menggunakan Java, sehingga kakas yang digunakan dalam membuat GUI haruslah berasal dari ekosistem bahasa Java (seperti `JavaFX`¹ atau `Swing`). Anda juga hanya diperbolehkan untuk membuat aplikasi bertipe *desktop* sehingga **tidak diperbolehkan untuk membuat aplikasi web**.

Semua fitur wajib (SPL, Determinan, Matriks Balikan, Interpolasi Polinomial, *Natural Cubic Spline*, dan Regresi Spline Kubik) harus dapat diakses dan dijalankan dengan baik melalui GUI jika Anda membuat bonus ini. Sangat disarankan agar GUI dapat menampilkan visualisasi langkah-langkah komputasi secara menarik (misalnya visualisasi matriks pada tahapan OBE, grafik kurva hasil interpolasi, maupun kurva hasil regresi) untuk meningkatkan pemahaman pengguna. Bonus ini bernilai maksimal **15 poin**.

3.2.2 Image Inpainting (Hole Filling) dengan Persamaan Laplace

Ini adalah penerapan dari penyelesaian SPL yang dapat merekonstruksi bagian citra yang rusak atau menghilangkan objek yang tidak diinginkan secara otomatis (*image inpainting* / *hole filling*). Berikut adalah contoh penerapannya,



Gambar 4: Contoh penerapan *image inpainting* pada restorasi citra rusak dan penghapusan teks

Teknik *image inpainting* bekerja dengan mengisi area yang hilang atau rusak dengan memanfaatkan informasi dari piksel-piksel sekitarnya yang masih utuh sedemikian rupa sehingga hasil rekonstruksinya tampak alami. Selengkapnya mengenai teknik ini dapat dibaca pada artikel [Image Inpainting](#).

Pada bonus ini, masukan terdiri atas sebuah berkas gambar asli dan sebuah gambar *mask* hitam-putih dengan ukuran piksel yang sama. Area berwarna **putih** (nilai 255) pada *mask* menandakan piksel yang rusak atau hilang dan perlu diestimasi nilainya, sedangkan area berwarna **hitam** (nilai 0) menandakan piksel yang masih utuh.

Nilai intensitas piksel yang hilang pada koordinat (i, j) dapat dimodelkan melalui diskritisasi Persamaan Laplace:

$$4f_{i,j} - f_{i-1,j} - f_{i+1,j} - f_{i,j-1} - f_{i,j+1} = 0$$

- $f_{i,j}$: nilai intensitas piksel yang dicari pada koordinat (i, j) .
- $f_{i-1,j}, f_{i+1,j}, f_{i,j-1}, f_{i,j+1}$: nilai intensitas piksel-piksel tetangga (atas, bawah, kiri, dan kanan). Jika piksel tetangga merupakan piksel utuh (berada di luar area *mask*), nilai intensitasnya telah diketahui sebagai syarat batas (*boundary condition*) dan dipindahkan ke ruas kanan persamaan.

¹Gunakan `Scene Builder` untuk membantu merancang antarmuka secara interaktif (*drag-and-drop*).

Operasi matriks untuk menyelesaikan sistem ini cukup intensif. Jika area putih pada citra *mask* memiliki N buah piksel, maka akan terbentuk SPL berukuran $N \times N$. Dengan nilai N yang besar, penyelesaian menggunakan eliminasi Gauss langsung memiliki kompleksitas $\mathcal{O}(N^3)$ yang terlalu lambat. Oleh karena itu, untuk menyelesaikan bonus ini, Anda **WAJIB** menggunakan metode penyelesaian iteratif, yakni **Metode Jacobi** atau **Metode Gauss-Seidel**.

Untuk mencegah komputasi yang terlalu berat, ukuran gambar dibatasi maksimal 512×512 piksel. Jika ukuran gambar masukan melebihi batas tersebut, program wajib menampilkan pesan peringatan dan menolak memproses gambar. Untuk keperluan membaca serta menyimpan berkas citra, Anda diperbolehkan menggunakan pustaka standar Java seperti `javax.imageio.ImageIO`.

Masukan untuk fitur ini terdiri atas:

- Berkas citra asli yang akan diperbaiki (format `.jpg` atau `.png`).
- Citra *mask* hitam-putih berdimensi sama dengan gambar asli sebagai penanda area kerusakan.

Luaran yang dihasilkan meliputi:

- Gambar hasil rekonstruksi yang dapat disimpan ke dalam format `.png` atau `.jpg`.
- Informasi proses komputasi pada layar, meliputi jumlah iterasi hingga konvergen, nilai galat (*error*) akhir, serta dimensi SPL yang diselesaikan.

Bonus ini bernilai maksimal **10 poin**.

3.2.3 Menggunakan $\text{\LaTeX}$ dalam Penulisan Laporan

$\text{\LaTeX}$ adalah sistem penyusunan dokumen yang sangat populer di ranah akademik dan ilmiah, khususnya untuk menyusun dokumen yang membutuhkan penulisan rumus matematika dan tabel secara rapi dan formal. Bahkan seluruh dokumen spesifikasi tugas besar ini pun disusun menggunakan $\text{\LaTeX}$.

Salah satu keuntungan utama $\text{\LaTeX}$ adalah berkas sumbernya yang berupa teks biasa (`.tex`), sehingga sangat cocok dan nyaman digunakan secara kolaboratif bersama Git untuk *version control*. Untuk mengklaim bonus ini, buatlah laporan tugas besar Anda menggunakan $\text{\LaTeX}$, lalu ekspor hasilnya ke format PDF. Pastikan seluruh berkas mentah (`.tex`) juga disertakan di dalam folder `doc/` sebagai bukti penggunaan $\text{\LaTeX}$. Kerapian struktur laporan, konsistensi format penulisan, dan keindahan penyusunan rumus matematika akan menjadi aspek utama yang dinilai. Bonus ini bernilai maksimal **5 poin**.

3.2.4 Video Penjelasan Program

Terdapat banyak cara untuk menjelaskan cara kerja sebuah program, namun penjelasan secara visual dan kreatif adalah teknik penjelasan yang paling menarik perhatian orang. Oleh karena itu, Anda diperbolehkan membuat sebuah video penjelasan mengenai program yang telah dibuat dengan durasi maksimal 15 menit.

Video ini minimal memuat pembahasan mengenai arsitektur dan alur program, demonstrasi langsung dari seluruh fitur wajib serta bonus yang dikerjakan, hingga penjelasan mengenai kendala teknis yang dihadapi beserta solusinya. Buatlah video ini sekreatif mungkin (jangan sekadar *screen recording* biasa, melainkan tunjukkan kemampuan presentasi, animasi visual, narasi yang informatif, atau bahkan keterampilan berakting kalian :D). Video dapat diunggah ke platform seperti YouTube dengan mode *unlisted* (tidak terdaftar publik), lalu sertakan tautannya di dalam laporan. Bonus ini bernilai maksimal **3 poin**.

3.3 Prosedur Pengerjaan

1. Tugas dikerjakan secara berkelompok dengan anggota sebanyak **3 atau 4 orang**. Anggota kelompok boleh berasal dari kelas yang berbeda selama terdaftar sebagai peserta mata kuliah IF2123 Aljabar Linier dan Geometri. Setiap kelompok wajib mendaftarkan anggotanya dan memilih asisten pembimbing melalui [sheets pendataan kelompok](#) sebelum **12 September 2026 pukul 09.00 WIB**. Peserta yang belum terdaftar setelah tenggat akan dipasangkan ke dalam kelompok secara acak.
2. Salah satu perwakilan kelompok bertindak sebagai pemilik repositori. Pemilik membuka *template* repositori [IRK-23/Tubes1-Algeo-2026](#), memilih opsi **Use this template**, kemudian membuat satu repositori baru pada akun GitHub pribadinya dengan ketentuan:

- Nama repositori adalah `algeo26-tb1-<nama-kelompok>`.
 - Repositori bersifat *private*.
 - Setiap kelompok hanya membuat satu repositori.
3. Pemilik repositori **wajib mengundang seluruh anggota kelompok dan asisten pembimbing pilihan** sebagai *collaborator*. Seluruh undangan harus diterima sebelum **13 September 2026 pukul 23.59 WIB**. Akses *collaborator* **tidak boleh** dicabut sampai seluruh proses penilaian tugas besar selesai.
 4. Setiap anggota kelompok wajib menggunakan akun GitHub masing-masing dalam pengerjaan tugas. Pembagian kontribusi anggota harus terlihat secara transparan melalui riwayat *commit* repositori.
 5. Pertanyaan seputar spesifikasi tugas besar dapat disampaikan melalui [sheets QnA](#).
 6. Kelompok dapat mengajukan asistensi kepada asisten pembimbing apabila diperlukan. Asisten pembimbing kelompok juga akan bertindak sebagai penguji pada saat demo.
 7. Setelah tenggat pengumpulan tugas besar, jadwal dan teknis pelaksanaan demo akan diumumkan melalui milis.
 8. **DILARANG** menyontek pekerjaan kelompok lain, melakukan plagiarisme pada naskah laporan maupun kode program, serta menggunakan bantuan *generative AI* (seperti ChatGPT, DeepSeek, Claude, Gemini, dan layanan sejenis), kecuali untuk penggunaan yang secara eksplisit diizinkan pada spesifikasi. Seluruh berkas kode dan laporan harus merupakan karya orisinal mandiri kelompok. Segala bentuk kecurangan dan pelanggaran integritas akademik akan ditindak tegas sesuai dengan sanksi dan ketentuan akademik yang berlaku di Institut Teknologi Bandung.

3.4 Spesifikasi Laporan

Setiap kelompok wajib menyusun laporan tugas besar yang mendokumentasikan seluruh proses pengerjaan, mulai dari analisis masalah, dasar teori, implementasi, hingga hasil eksperimen. Laporan ditulis dalam bahasa Indonesia yang baik dan benar, tidak perlu terlalu panjang, tetapi harus tepat sasaran, jelas, dan informatif.

3.4.1 Struktur Laporan

Laporan tugas besar minimal harus terdiri dari bagian-bagian berikut:

1. **Cover (Sampul)**
Berisi judul laporan, kode dan nama mata kuliah, nama dan NIM anggota kelompok, serta foto bersama anggota kelompok. Foto ini menggantikan logo institusi (logo “gajah” Ganesha) pada sampul.
2. **Bab 1: Pendahuluan**
Berisi deskripsi masalah secara umum, latar belakang pengerjaan tugas besar, dan tujuan yang ingin dicapai. Bab ini dapat menyalin dari bagian Pendahuluan pada dokumen spesifikasi ini.
3. **Bab 2: Dasar Teori**
Berisi penjelasan teori singkat mengenai semua metode yang diimplementasikan, meliputi:
 - a. Sistem Persamaan Linier (Eliminasi Gauss, Eliminasi Gauss-Jordan, Metode Matriks Balikan, dan Kaidah Cramer).
 - b. Determinan Matriks (Reduksi Baris/OBE dan Ekspansi Kofaktor).
 - c. Matriks Balikan (Metode Augmentasi dan Metode Adjoin).
 - d. Interpolasi Polinomial.
 - e. Natural Cubic Spline Interpolation (sertakan penjelasan tentang sistem tridiagonal dan syarat batas natural).
 - f. Regresi Spline Kubik (sertakan penjelasan tentang *truncated power basis* dan Persamaan Normal).

- g. Jika mengerjakan bonus Image Inpainting, tambahkan teori tentang diskritisasi Persamaan Laplace dan Metode Jacobi/Gauss-Seidel.

4. Bab 3: Implementasi

Berisi penjelasan mengenai implementasi pustaka dan program dalam Java, meliputi:

- Struktur kelas-kelas yang didefinisikan (atribut dan metode penting).
- Arsitektur pustaka (bagaimana modul-modul saling terhubung).
- Alur program secara garis besar (misalnya: diagram alir atau deskripsi langkah-langkah utama).
- Penjelasan tentang penggunaan Maven sebagai *build tool* dan cara pembuatan berkas `.jar`.

5. Bab 4: Eksperimen

- Berisi hasil eksekusi program terhadap seluruh kasus uji yang tercantum pada Bab 5 (Studi dan Kasus Uji) di dokumen spesifikasi ini.
- Setiap kasus uji harus disertai dengan:
 - Input yang diberikan.
 - Output yang dihasilkan oleh program.
 - Analisis terhadap hasil yang diperoleh (misalnya: apakah sesuai dengan solusi eksak, bagaimana performa metode, atau perbandingan antarmetode).
- Jika terdapat perbedaan hasil antarmetode (misal: karena galat pembulatan), berikan analisis mengenai penyebabnya.

6. Bab 5: Penutup

Memuat ulasan penutup laporan yang mencakup:

- Kesimpulan atas seluruh capaian tugas besar serta evaluasi kelebihan dan kekurangan sistem yang telah dibangun.
- Saran relevan untuk perbaikan, optimasi performa, atau pengembangan fitur di masa mendatang.
- Refleksi pengalaman setiap anggota kelompok mengenai proses pengerjaan, wawasan baru yang didapatkan, serta dinamika pemecahan kendala.

7. Daftar Pustaka

- Berisi seluruh referensi (buku, jurnal, artikel web, atau sumber lain) yang digunakan selama pengerjaan tugas besar.
- Gunakan **format sitasi IEEE** (misal: [1] H. Anton, *Elementary Linear Algebra*, 10th ed. Hoboken, NJ: Wiley, 2010.).

8. Lampiran

- Berisi tautan menuju **repository GitHub** kelompok.
- Jika mengerjakan bonus Video, cantumkan tautan video YouTube.
- Jika mengerjakan bonus $\text{\LaTeX}$, cantumkan tautan unduhan atau lokasi berkas `.tex` di repository.
- Pernyataan integritas akademik yang menegaskan bahwa seluruh pekerjaan tugas besar diselesaikan secara mandiri tanpa pemanfaatan kecerdasan buatan generatif (*Generative AI*). Setiap anggota kelompok wajib mencantumkan tanda tangan beserta nama lengkap pada lembar pernyataan ini di dalam lampiran laporan. Template pernyataan dapat mengikuti contoh berikut:

Tugas ini disusun sepenuhnya tanpa bantuan kecerdasan buatan (*Generative AI*), melainkan hasil pemikiran dan analisis mandiri.

[Tanda tangan mahasiswa]
[Nama mahasiswa]

[Tanda tangan mahasiswa]
[Nama mahasiswa]

[Tanda tangan mahasiswa]
[Nama mahasiswa]

3.4.2 Ketentuan Penulisan dan Format

1. Laporan wajib disusun menggunakan bahasa Indonesia yang baik, lugas, dan benar sesuai dengan pedoman Ejaan Bahasa Indonesia (EYD).
2. Panjang laporan tidak dibatasi oleh jumlah halaman tertentu, namun harus memuat seluruh komponen pada struktur laporan secara ringkas dan padat. Penilaian lebih menitikberatkan pada kedalaman analisis serta kejelasan isi dibanding banyaknya halaman.
3. Berkas laporan dikumpulkan dalam format PDF. Bagi kelompok yang mengambil bonus penulisan laporan dengan $\text{\LaTeX}$, sertakan pula seluruh berkas sumbernya (`.tex` dan aset pelengkap) di dalam direktori `doc/`.
4. Laporan diatur menggunakan kertas berukuran A4 dengan margin standar (atas 2.5 cm, bawah 2.5 cm, kiri 3 cm, kanan 2.5 cm), ukuran fon 12 pt (menggunakan fon standar seperti Times New Roman, Arial, atau Calibri), serta menyertakan nomor halaman yang konsisten.
5. Setiap lembar laporan hendaknya mencantumkan identitas yang jelas pada bagian kepala atau kaki halaman (*header/footer*), seperti nama kelompok dan kode mata kuliah (IF2123), agar susunan halaman tetap mudah diidentifikasi.
6. Kode sumber program (*source code*) tidak perlu disertakan di dalam naskah laporan. Pemeriksaan implementasi kode dilakukan langsung melalui tautan repositori GitHub yang dilampirkan.

3.5 Kriteria Penilaian

Penilaian tugas besar ini dirincikan pada tabel di bawah. Nilai terdiri atas tiga komponen utama, yaitu implementasi dan demo program (80 poin), laporan (20 poin), dan bonus. Total poin pada kolom bonus memang berjumlah 33, tetapi skor bonus yang dihitung maksimal 20 poin. Artinya, jika kalian mengerjakan semua bonus dan memperoleh nilai lebih dari 20, maka skor bonus akan tetap dicatat sebagai 20 (tidak bisa melebihi 20 poin).

Tabel 1: Rincian Skor Maksimum untuk Tiap Komponen Penilaian

Komponen	Skor
Implementasi dan Demo	80
Dokumentasi	20
(Bonus) Antarmuka Pengguna Grafis (GUI)	15
(Bonus) Image Inpainting (Hole Filling) dengan Persamaan Laplace	10
(Bonus) Menggunakan $\text{\LaTeX}$ dalam Penulisan Laporan	5
(Bonus) Video Penjelasan Program	3
Total	120

4 Pengumpulan

4.1 Deliverables

Berkas yang dikumpulkan oleh setiap kelompok terdiri atas kode sumber (*source code*), program siap dijalankan, data kasus uji, dan laporan tugas besar dengan ketentuan sebagai berikut.

1. Program dan laporan disimpan di dalam repositori GitHub *private* yang dibuat dari repositori *template* resmi.
2. Repositori sekurang-kurangnya memuat empat folder berikut:
 - a. Folder `docs` yang berisi laporan dalam format PDF dengan nama `<nama-kelompok>.pdf`.
 - b. Folder `bin` yang berisi berkas `JAR` dari program yang dapat dijalankan.
 - c. Folder `src` yang berisi seluruh kode sumber program.
 - d. Folder `test` yang berisi data kasus uji dalam berkas `.txt` serta berkas pendukung pengujian lainnya.
3. Setiap kelompok wajib menggunakan konvensi *semantic commit message* dan menjelaskan perubahan yang dilakukan.
4. Repositori wajib memiliki berkas `README.md` yang sekurang-kurangnya memuat penjelasan singkat program, alur program, dan tata cara menjalankan program. Gunakan *README ini* sebagai contoh.
5. Pastikan repositori bersifat *private*² selama pengerjaan tugas besar.
6. Pastikan *bytecode* di dalam folder `bin` dapat dijalankan sesuai dengan petunjuk pada `README.md`.
7. Selain dijalankan saat demo, program akan dicoba dijalankan asisten di luar waktu demo untuk salah satu aspek penilaian. Asisten pemeriksa tidak akan melakukan pengaturan atau kompilasi lagi agar program dapat berjalan. Program yang tidak dapat dijalankan tidak akan diberi nilai.
8. Apabila kelompok mengerjakan fitur bonus, seluruh kode sumber, berkas pendukung, dan petunjuk penggunaan fitur tersebut harus disertakan di dalam repositori. Tautan video untuk bonus video dicantumkan pada `README.md`.

4.2 Teknis Pengumpulan

1. Program dan laporan dikumpulkan dengan membuat *rilis* pada repositori GitHub untuk tiap kelompok. Sebelum membuat rilis, pastikan seluruh *deliverables* telah tersedia dan tag rilis mengacu pada *commit* terakhir yang akan dinilai.
2. Tag rilis menggunakan format `v1.x`, dengan x merupakan nomor revisi ($x = 0$ untuk rilis pertama). Judul rilis menggunakan format `v1.x-<nama-kelompok>`. Deskripsi rilis sekurang-kurangnya memuat nama kelompok, daftar anggota, dan ringkasan perubahan apabila rilis tersebut merupakan revisi.
3. Pengumpulan dilakukan melalui *Google Form* berikut. Pengumpulan cukup dilakukan satu kali oleh perwakilan kelompok. Setiap kelompok wajib memastikan bahwa tautan rilis dapat diakses oleh asisten pembimbing dan koordinator asisten. Kesalahan tautan atau akses repositori menjadi **tanggung jawab kelompok**.
4. Kelompok **boleh melakukan revisi** dengan membuat rilis baru dan mengirimkan kembali *Google Form*. Versi yang digunakan untuk penilaian adalah rilis terakhir yang dikumpulkan melalui *Google Form*. Perubahan pada repositori yang tidak disertai pembuatan dan pengumpulan rilis baru tidak akan dinilai.
5. Waktu pengumpulan ditentukan berdasarkan **waktu pengiriman Google Form**. Rilis tidak boleh diubah setelah tautannya dikumpulkan. Apabila diperlukan perubahan, kelompok wajib membuat nomor revisi baru dan mengirimkan kembali formulir. Tenggat pengumpulan tugas besar adalah hari **Rabu, 30 September 2026 pukul 21.23 WIB**.

²Bisa diubah ke *public* setelah tenggat pengumpulan tugas besar berlalu.

6. Keterlambatan dikenai penalti sebesar **1 poin untuk setiap 6 menit keterlambatan**. Durasi keterlambatan dihitung dari selisih antara tenggat pengumpulan dan waktu pengiriman terakhir pada Google Form, kemudian dibulatkan ke atas ke kelipatan 6 menit.³

³Misalnya, keterlambatan 3 menit maupun 6 menit sama-sama menghasilkan penalti 1 poin.

5 Studi dan Kasus Uji

Untuk menguji program yang telah dibuat, lakukan pengujian terhadap beberapa matriks dan persoalan berikut.

5.1 Determinan Matriks

Tentukan nilai determinan dari matriks-matriks berikut. Uji setiap kasus menggunakan kedua metode perhitungan determinan yang terdapat pada Bagian 3.1.2

5.1.1 Kasus Uji 1

$$\begin{bmatrix} 0 & 2 & -1 & 3 \\ 4 & 1 & 2 & -2 \\ -2 & 3 & 0 & 1 \\ 1 & -1 & 5 & 2 \end{bmatrix}$$

5.1.2 Kasus Uji 2

$$\begin{bmatrix} 2 & -1 & 3 & 0 & 4 \\ 1 & 2 & -2 & 5 & -1 \\ 3 & 1 & 1 & 2 & 0 \\ 0 & 4 & -1 & 3 & 2 \\ 5 & 0 & 4 & 2 & 4 \end{bmatrix}$$

5.1.3 Kasus Uji 3

$$\begin{bmatrix} \frac{1}{2} & 1 & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{5}{4} & \frac{1}{4} & \frac{1}{2} \\ -\frac{1}{10} & -\frac{2}{5} & \frac{1}{5} & -\frac{3}{10} \\ \frac{3}{5} & \frac{5}{4} & \frac{9}{20} & \frac{3}{5} \end{bmatrix}$$

5.2 Invers Matriks

Tentukan invers dari matriks-matriks berikut. Uji setiap kasus menggunakan kedua metode pencarian invers yang terdapat pada Bagian 3.1.2. Jika matriks tidak memiliki invers, program harus menampilkan pesan yang sesuai.

5.2.1 Kasus Uji 1

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

5.2.2 Kasus Uji 2

$$\begin{bmatrix} 1 & 2 & 0 & -1 & 3 \\ 2 & -1 & 4 & 0 & 1 \\ 0 & 3 & 1 & 2 & -2 \\ 4 & 0 & 5 & 1 & 2 \\ 3 & 1 & 4 & -1 & 4 \end{bmatrix}$$

5.3 Sistem Persamaan Linier ($Ax = b$)

Tentukan solusi dari setiap SPL ($Ax = b$) berikut. Uji setiap kasus menggunakan seluruh metode penyelesaian SPL pada Bagian 3.1.2. Jika suatu metode tidak dapat digunakan, program harus menampilkan pesan yang sesuai.

5.3.1 Kasus Uji 1

$$A = \begin{bmatrix} 0 & 2 & -1 & 1 \\ 1 & -1 & 2 & 3 \\ 2 & 1 & 1 & -1 \\ -1 & 3 & 0 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} -5 \\ 15 \\ 1 \\ -3 \end{bmatrix}$$

5.3.2 Kasus Uji 2

$$A = \begin{bmatrix} 1 & 2 & -1 & 0 & 3 \\ 2 & 4 & 1 & -1 & 5 \\ -1 & -2 & 2 & 3 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ 10 \\ -1 \end{bmatrix}$$

5.3.3 Kasus Uji 3

$$A = \begin{bmatrix} 1 & -2 & 3 & 1 \\ 2 & 1 & -1 & 4 \\ 0 & 3 & 2 & -2 \\ 3 & -1 & 2 & 5 \end{bmatrix}, \quad b = \begin{bmatrix} 4 \\ -1 \\ 3 \\ 4 \end{bmatrix}$$

5.3.4 Kasus Uji 4

Diberikan matriks *discrete Laplacian* satu dimensi berukuran $(n \times n)$:

$$A_n = \begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \ddots & \vdots \\ 0 & -1 & 2 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \cdots & 0 & -1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}$$

Elemen-elemen matriks A_n didefinisikan sebagai:

$$(A_n)_{ij} = \begin{cases} 2, & i = j, \\ -1, & |i - j| = 1, \\ 0, & \text{lainnya.} \end{cases}$$

Selesaikan SPL untuk $n = 6$ dan $n = 10$.

5.4 Sistem Persamaan Linier Berbentuk Matriks Augmented

Tentukan solusi dari setiap SPL yang direpresentasikan oleh matriks augmented berikut. Uji setiap kasus menggunakan seluruh metode penyelesaian SPL pada Bagian 3.1.2.

5.4.1 Kasus Uji 1

$$\left[\begin{array}{cccc|c} 0 & 1 & -1 & 2 & -1 \\ 2 & 0 & 1 & -1 & 3 \\ -1 & 2 & 0 & 1 & -3 \\ 1 & -1 & 2 & 0 & 9 \end{array} \right]$$

5.4.2 Kasus Uji 2

$$\left[\begin{array}{cccccc|c} 1 & 0 & 2 & -1 & 0 & 3 & -7 \\ 0 & 1 & -1 & 2 & 1 & 0 & 6 \\ 2 & -1 & 0 & 1 & 3 & -2 & 13 \\ 1 & 1 & 1 & 1 & 1 & 1 & 3 \end{array} \right]$$

5.5 Sistem Persamaan Linier Berbentuk Umum

5.5.1 Kasus Uji 1

Tentukan seluruh solusi sistem persamaan

$$\begin{aligned} 3x_1 + 2x_2 - x_3 + 4x_4 &= 5 \\ x_1 - 3x_2 + 2x_3 + x_4 &= -2 \\ 4x_1 + x_2 + 5x_3 - 2x_4 &= 7 \\ 2x_1 - x_2 + 3x_3 + x_4 &= 3 \end{aligned}$$

5.5.2 Kasus Uji 2

Tentukan seluruh solusi sistem persamaan

$$\begin{aligned} x_1 + x_2 + x_3 &= 10.00 \\ x_4 + x_5 + x_6 &= 12.00 \\ x_7 + x_8 + x_9 &= 9.00 \\ 0.03521(x_1 + x_4 + x_7) + 0.68(x_2 + x_6) + 0.55432x_9 &= 8.42 \\ 0.87654(x_1 + x_4 + x_7) + 0.30(x_3 + x_5 + x_8) &= 11.05 \\ 0.03521(x_1 + x_4 + x_7) + 0.68(x_3 + x_5) + 0.55432x_2 &= 6.19 \\ x_1 + x_5 + x_9 &= 11.00 \\ x_2 + x_4 + x_8 &= 8.50 \\ x_3 + x_6 + x_7 &= 11.50 \end{aligned}$$

5.6 Aplikasi Sistem Persamaan Linear

Diberikan dataset berukuran kecil yang berisi informasi tentang luas rumah dalam ratusan m² dan harga dalam ratusan juta

X (luas rumah)	Y (harga)
1	3
2	5
3	6

Misalkan kita ingin membuat sebuah model prediksi harga rumah terhadap luas rumah berbentuk $y = \theta_0 + \theta_1 x$. Sekarang, perhatikan juga bahwa matriks-matriks yang bersesuaian dengan model yang ingin kita bentuk memiliki persamaan

$$Y = \theta X$$

Kita akan mencari parameter θ_0 dan θ_1 menggunakan *normal equation*

$$\theta = (X^T X)^{-1} X^T Y$$

Dengan demikian, kita mendapatkan sistem persamaan:

$$\begin{aligned} 3\theta_0 + (x_1 + x_2 + x_3)\theta_1 &= y_1 + y_2 + y_3 \\ (x_1 + x_2 + x_3)\theta_0 + (x_1^2 + x_2^2 + x_3^2)\theta_1 &= x_1 y_1 + x_2 y_2 + x_3 y_3 \end{aligned}$$

Tentukan nilai θ_0 dan θ_1 yang memenuhi persamaan linear tersebut.

5.7 Interpolasi Polinomial

a. Studi Kasus 1

Gunakan tabel di bawah ini untuk mencari polinomial interpolasi yang melewati titik - titik di bawah ini :

x	0.2	0.4	0.6	0.8	1.0	1.2	1.4
$f(x)$	0.182	0.336	0.470	0.588	0.693	0.788	0.875

Kemudian, tentukan juga nilai - nilai dari $f(x)$ dari polinomial yang telah dikonstruksi untuk nilai - nilai berikut :

$$x = 0.3$$

$$x = 0.55$$

$$x = 0.95$$

$$x = 1.38$$

b. Studi Kasus 2

Seorang insinyur sedang melakukan pengujian terhadap sebuah komponen elektronik untuk mengukur bagaimana suhu komponen meningkat seiring bertambahnya waktu ketika dialiri oleh arus listrik terus - menerus. Data suhu (T dalam $^{\circ}\text{C}$) yang tercatat pada beberapa waktu pengukuran (t , dalam menit) adalah sebagai berikut :

t (menit)	0	2	4	6	8	10
T	25.0	31.2	38.6	47.5	58.1	70.8

Insinyur tersebut ingin mengetahui perkiraan suhu komponen pada waktu-waktu tertentu di antara titik pengukuran, tanpa harus melakukan pengujian ulang. Buatlah program yang menerima masukan nilai t dan menghitung perkiraan nilai T menggunakan interpolasi polinomial.

Lakukan pengujian pada nilai - nilai berikut

$$t = 1 \quad T = ?$$

$$t = 3.5 \quad T = ?$$

$$t = 7 \quad T = ?$$

$$t = 9.2 \quad T = ?$$

Jika komponen rusak pada suhu 65°C , tentukan di menit ke berapa komponen rusak.

5.8 Studi Interpolasi Natural Cubic Spline

Konstruksikan kurva *natural cubic spline* yang melewati 5 titik data berikut.

i	0	1	2	3	4
x_i	1	2	3	4	5
y_i	2	3	4	5	6

5.9 Studi Kasus Regresi Spline

Konstruksikan kurva spline linear yang secara tepat melalui 3 titik data : $(1, 2)$, $(3, 6)$, dan $(5, 4)$.

6 Referensi

6.1 SPL, Determinan, dan Matriks

- H. Anton and C. Rorres, *Elementary Linear Algebra: Applications Version*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, 2010. [Online]. Available: <https://archive.org/details/ElementryLinearAlgebraByHowardAnton10thEdition>

6.2 Interpolasi Polinomial

- R. L. Burden and J. D. Faires, *Numerical Analysis*, 9th ed. Boston, MA, USA: Brooks/Cole, Cengage Learning, 2011, ch. 3 (Interpolation and Polynomial Approximation). [Online]. Available: https://faculty.ksu.edu.sa/sites/default/files/numerical_analysis_9th.pdf
- G. Muntingh, *Topics in Polynomial Interpolation Theory*, Ph.D. dissertation, Centre of Mathematics for Applications, University of Oslo, Oslo, Norway, Dec. 2010. [Online]. Available: https://www.researchgate.net/publication/265819309_Topics_in_Polynomial_Interpolation_Theory

6.3 Natural Cubic Spline Interpolation

- R. L. Burden and J. D. Faires, *Numerical Analysis*, 9th ed. Boston, MA, USA: Brooks/Cole, Cengage Learning, 2011, ch. 3, sec. 3.5 (Cubic Splines). [Online]. Available: https://faculty.ksu.edu.sa/sites/default/files/numerical_analysis_9th.pdf
- C. de Boor, *A Practical Guide to Splines*, rev. ed. New York, NY, USA: Springer-Verlag, 2001. [Online]. Available: <https://doi.org/10.1007/978-1-4612-6333-3>

6.4 Regresi Spline (*Regression Splines*)

- T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009, ch. 5 (Basis Expansions and Regularization). [Online]. Available: <https://course.ccs.neu.edu/ds4420sp20/readings/ESLII.pdf>
- G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*, 2nd ed. New York, NY, USA: Springer, 2021, ch. 7 (Moving Beyond Linearity). [Online]. Available: <https://www.statlearning.com/>

6.5 Image Inpainting dengan Persamaan Laplace

- M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image Inpainting," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '00)*, New York, NY, USA: ACM, 2000, pp. 417–424. [Online]. Available: <https://doi.org/10.1145/344779.344972>
- P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3, pp. 313–318, 2003. [Online]. Available: <https://doi.org/10.1145/882262.882269>

7 Surat Cinta dari Asisten

Komentar 7.1: Ares

BUKTIKAN BAHWA KALIAN TANGGUH DALAM TUBESKU, SEMANGATTT

Komentar 7.2: Dzaki



Semangat Tugas besar pertama kalian, ini permulaan untuk tugas besar yang lain!

Komentar 7.3: Daniel

jangan dione-shot yak

Komentar 7.4: An-Dafa



If you know, you know

Komentar 7.5: Manuel

ad astra per aspera.

Komentar 7.6: Jonathan

ibarat ini baru permulaan