

LAPORAN TUGAS BESAR I
IF 2123 ALJABAR LINEAR DAN GEOMETRI
Sistem Persamaan Linier, Determinan, dan Aplikasinya



Disusun oleh:

Bryan Cornelius Lauwrence	(13522033)
Venantius Sean Ardi Nugroho	(13522078)
Julian Chandra Sutadi	(13522080)

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2023

BAB I

DESKRIPSI MASALAH

A. Latar Belakang

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).

SPL tersebut dapat digunakan dalam pencarian persamaan dengan interpolasi polinomial, regresi linear berganda, dan interpolasi bikubik. Interpolasi dan regresi tersebut digunakan untuk mengestimasi persamaan garis maupun persamaan bidang. Dengan mengolah beberapa titik sehingga dihasilkan sebuah matrik. Matriks tersebut kemudian diperhitungkan sebagai SPL agar diperoleh persamaan hasil estimasi tersebut.

Oleh karena itu, kami bertujuan membuat program yang dapat mengimplementasikan SPL tersebut dengan metode-metode yang ada. Kami juga membuat program untuk menghitung persamaan hasil interpolasi dan regresi sekaligus menghitung data tertentu. Program kami dibuat dengan bahasa pemrograman Java dengan implementasi kelas-kelas yang diperlukan.

B. Tujuan

Tujuan dari tugas besar ini adalah sebagai berikut:

1. Menghitung solusi SPL dengan bermacam-macam metode.
2. Mengestimasi persamaan dari beberapa titik dengan interpolasi atau regresi.
3. Mencari hasil nilai dari titik yang diberikan pada interpolasi dan regresi.

BAB II

TEORI SINGKAT

A. Determinan Matriks

Determinan matriks berukuran 2×2 didefinisikan sebagai berikut:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\det(A) = ad - bc,$$

dengan $\det(A)$ merupakan determinan matriks A. Sedangkan determinan matriks 3×3 didefinisikan sebagai berikut:

$$A = \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{bmatrix}$$

$$\det(A) = a_1a_5a_9 + a_2a_6a_7 + a_3a_4a_8 - a_3a_5a_7 - a_1a_6a_8 - a_2a_4a_9$$

Namun, rumus tersebut berlaku dengan cara yang berbeda pada matriks persegi berukuran $n \times n$.

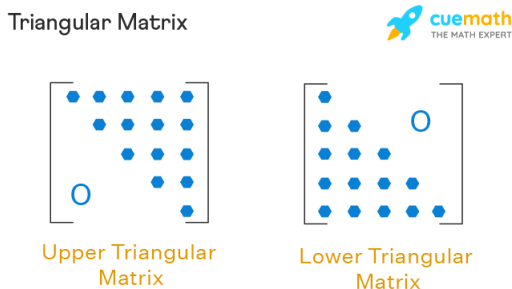
Pada matriks persegi berukuran $n \times n$, determinan matriks dihitung dengan metode lain yang lebih mudah, yaitu teknik reduksi baris (akan dibahas lebih lanjut pada subbab selanjutnya). Teknik reduksi baris bertujuan menghasilkan matriks segitiga atas atau matriks segitiga bawah. Determinan dari matriks segitiga atas maupun matriks segitiga bawah merupakan hasil perkalian elemen-elemen diagonalnya. Namun, implementasi reduksi baris akan merubah determinan suatu matriks dengan ketentuan sebagai berikut:

- Sebuah baris di matriks A dikalikan dengan sebuah konstanta menghasilkan matriks B, maka $\det(B) = k \det(A)$.
- Dua buah baris di matriks A ditukarkan dan menghasilkan matriks B, maka $\det(B) = -\det(A)$.
- Sebuah baris di matriks A ditambahkan dengan k kali baris lainnya menghasilkan matriks B, maka $\det(B) = \det(A)$.

Selain cara tersebut, determinan matriks $n \times n$ juga dapat diperoleh dengan metode kofaktor.

B. Teknik Reduksi Baris

Untuk menggunakan teknik ini, perlu diketahui mengenai Matriks Segitiga. Matrix segitiga adalah sebuah matrik persegi yang elemen bawah dan/atau atas diagonalnya adalah 0 semua. Matrik segitiga yang semua elemen diatas diagonal utamanya adalah 0 adalah matriks segitiga atas, sedangkan yang elemen di bawah diagonal utamanya adalah 0 adalah matriks segitiga bawah.



Gambar 1. Jenis Matriks Segitiga

Satu sifat yang bisa dimanfaatkan dari matriks segitiga adalah determinannya dapat dicari dengan cara mengalikan semua elemen dalam diagonal utamanya. Dengan memanfaatkan Operasi baris elementer dan sifat tersebut, bisa didapatkan determinan untuk sembarang matriks. Perlu diingat bahwa untuk mengubah suatu matriks sembarang menjadi matriks segitiga, terdapat penyesuaian determinan sebagai berikut :

Relationship	Operation
$\begin{vmatrix} ka_{11} & ka_{12} & ka_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = k \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$ $\det(B) = k \det(A)$	The first row of A is multiplied by k.
$\begin{vmatrix} a_{21} & a_{22} & a_{23} \\ a_{11} & a_{12} & a_{13} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = - \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$ $\det(B) = -\det(A)$	The first and second rows of A are interchanged.
$\begin{vmatrix} a_{11} + ka_{21} & a_{12} + ka_{22} & a_{13} + ka_{23} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$ $\det(B) = \det(A)$	A multiple of the second row of A is added to the first row.

Gambar 2, Penyesuaian determinan terhadap OBE

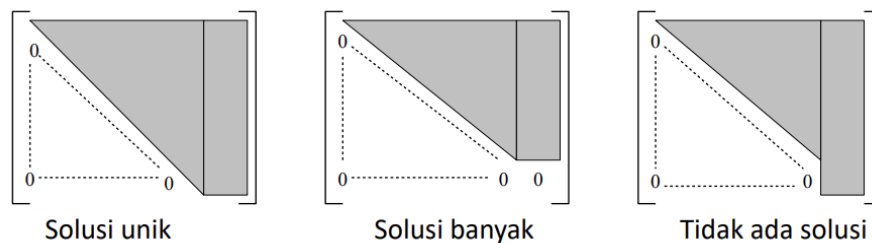
C. Metode Eliminasi Gauss

Metode eliminasi Gauss adalah salah satu metode yang dapat digunakan untuk mencari solusi dari sistem persamaan linier. Pertama - tama sistem persamaan linier dimasukkan ke dalam matriks augmented. Matriks tersebut akan disunting menggunakan operasi baris elementer sehingga terbentuk matriks eselon. Operasi baris elementer di sini terdiri dari :

- Mengalikan sebuah baris dengan konstanta tidak nol
- Menukarkan dua buah baris.
- Menambahkan sebuah baris dengan kelipatan baris lainnya.

Setelah matriks eselon ditemukan, kita bisa melakukan backward substitution untuk mendapatkan solusinya.

Perlu diingat bahwa bisa terdapat 3 kemungkinan solusi dari sistem persamaan linier, yaitu solusi unik, tidak memiliki solusi, dan solusi parametrik. Ciri - ciri dari solusi apa yang akan dihasilkan dari sebuah SPL dapat dilihat dari hasil matriks eselonnya. Bila terdapat baris yang berisi 0 semua, maka dapat dipastikan bahwa SPL tersebut memiliki hasil parametrik. Bila terdapat baris yang elemen pada kolom terakhirnya merupakan bilangan bukan 0 sedangkan elemen lainnya pada baris tersebut adalah 0, maka SPL tersebut tidak memiliki solusi. Selain kedua kondisi tersebut, maka solusinya unik.



Gambar 3, tiga kemungkinan SPL

Source : <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/Algeo-04-Tiga-Kemungkinan-Solusi-SPL-2023.pdf>

D. Metode Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan memiliki langkah yang mirip dengan eliminasi Gauss-Jordan, hingga mendapatkan matriks eselon. Bedanya, pada eliminasi Gauss-Jordan, matriks eselon tersebut akan diolah lebih lanjut menjadi matriks eselon tereduksi. Keuntungan dari pengolahan tersebut adalah bisa ditemukannya solusi non trivial dari matriks augmented yang homogen (semua elemen kolom terakhir dari matriks augmented adalah 0).

E. Matriks Balikan

Matriks balikan (*inverse*) adalah suatu matriks yang jika dikalikan dengan matriks asalnya, akan menghasilkan matriks identitas. Namun, tidak semua matriks memiliki suatu balikan. Matriks yang tidak memiliki balikan dinamakan matriks singular. Syarat suatu

matriks memiliki balikan, yaitu nilai determinannya bukan 0 dan matriks tersebut adalah matriks persegi (matriks yang jumlah baris dan kolomnya sama).

Untuk mencari balikan dari suatu matriks 2×2 , misalkan matriks A berukuran 2×2 sebagai berikut:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix},$$

maka invers matriks A (A^{-1}) diperoleh dengan cara berikut:

$$\frac{1}{\det(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Mencari matriks balikan dari matriks persegi berukuran $n \times n$, dilakukan dengan cara yang berbeda. Mencari invers matriks persegi ukuran sembarang dapat diperoleh dengan dua cara, yaitu metode eliminasi Gauss-Jordan dan metode mencari matriks balikan dengan matriks adjoin. Matriks adjoin adalah transpose dari matriks kofaktor (matriks kofaktor akan dibahas pada subbab selanjutnya).

Metode eliminasi Gauss-Jordan, suatu matriks akan digabungkan dengan matriks identitas, lalu matriks gabungan dieliminasi sehingga matriks awal menjadi matriks identitas dan matriks identitas menjadi matriks balikannya. Metode tersebut digambarkan sebagai berikut:

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

Sedangkan metode matriks adjoin, mencari matriks balikan dengan cara berikut:

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

dengan $\text{adj}(A)$ adalah matriks adjoin dari A .

Matriks balikan dapat digunakan untuk menghitung solusi dari SPL. Misalkan SPL berupa $Ax = b$, maka x diperoleh dengan cara $x = A^{-1}b$. SPL yang bisa diselesaikan dengan cara matriks balikan adalah matriks SPL yang solusinya unik.

F. Matriks Kofaktor

Setiap elemen matriks pada suatu matriks persegi memiliki nilai minor entrinya masing-masing. Minor entri dari elemen matriks baris ke- i dan kolom ke- j (M_{ij}) merupakan determinan dari matriks tersebut yang tidak melibatkan baris ke- i dan kolom ke- j . Melalui minor entri, kita dapat mencari nilai kofaktor dari elemen matriks tersebut. Kofaktor dari elemen matriks baris ke- i dan kolom ke- j (C_{ij}) adalah $((-1)^{i+j} M_{ij})$.

Determinan suatu matriks persegi juga dapat dicari dengan nilai kofaktor. Caranya adalah menjumlahkan seluruh elemen dari satu baris atau kolom yang setiap elemennya dikalikan dengan kofaktornya. Baris dan kolom yang dipilih bebas, selama penjumlahan dilakukan hanya pada baris atau kolom tersebut.

G. Kaidah Cramer

Kaidah Cramer adalah teknik untuk mencari solusi dari persamaan linear dengan memanfaatkan determinan. Misalkan diberikan persamaan dengan bentuk berikut,

$$\begin{aligned} a_{11}x_1 + \dots + a_{1n}x_n &= b_1 \\ a_{n1}x_1 + \dots + a_{nn}x_n &= b_n \end{aligned}$$

maka solusi dari persamaan tersebut bisa didapatkan dengan rumus

$$x_i = \det B_i / \det A; \quad i = 1, 2, \dots, n$$

dengan A adalah matriks koefisien dan B_i adalah matriks yang didapatkan dengan menggantikan kolom dengan indeks i dari matriks A dengan matriks kolom B .

H. Interpolasi Polinomial

Interpolasi polinomial adalah teknik yang digunakan untuk mengaproksimasi sebuah fungsi dengan cara menyesuaikan sebuah polinomial yang melalui beberapa titik yang diberikan. Tentu, karena ini adalah interpolasi polinomial, maka hasilnya adalah sebuah fungsi polinomial.

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

Dengan diberikan $n+1$ buah titik.

Dengan mensubstitusikan nilai x dan y dengan titik yang telah kita berikan dan mencari solusi SPL-nya, maka kita bisa mendapatkan nilai setiap nilai a pada persamaan polinom tersebut.

I. Regresi Linear Berganda

Regresi Linear 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.

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

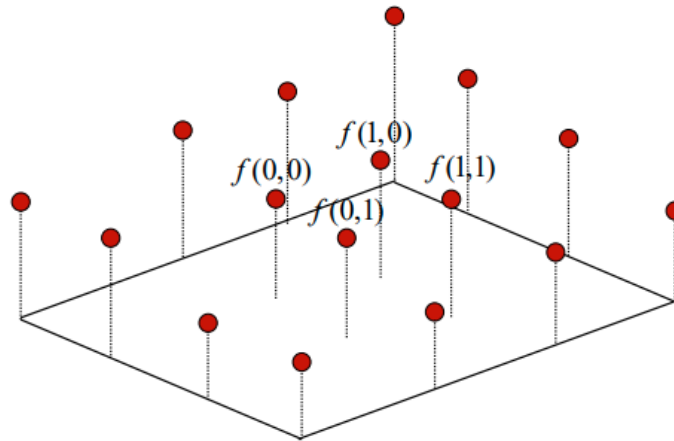
Untuk mendapatkan nilai dari setiap β_i dapat digunakan *Normal Estimation Equation for Multiple Linear Regression* sebagai berikut:

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

Kemudian, SPL tersebut dapat diselesaikan dengan metode-metode yang sudah disebutkan sebelumnya.

J. Interpolasi *Bicubic-Spline*

Interpolasi adalah teknik untuk mengaproksimasi fungsi yang dapat dibentuk dari kumpulan titik-titik beserta nilai fungsinya. Interpolasi dapat dilakukan untuk input satu dimensi maupun dua dimensi (yaitu, fungsi tiga dimensi). Pada interpolasi dua dimensi, ada tiga metode umum, yaitu interpolasi bilinear, bicubic, dan bicubic spline.



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

Interpolasi lebih baik jika fungsi aproksimasi yang dihasilkan lebih mulus. Hal ini yang menjadi tujuan dari pemanfaatan turunan dalam rumus mencari fungsi aproksimasi dari interpolasi *bicubic-spline*. Untuk mencari aproksimasi dari suatu titik yang berada di antara 4 titik, digunakan 12 titik yang mengelilingi 4 titik tersebut. Dari ke-16 titik titik ini terbentuk persamaan aproksimasi nilai fungsi di antara 4 titik dengan 16 suku. Nilai aproksimasi didapatkan dengan menyulihkan titik yang ingin diaproksimasi ke dalam fungsi hasil interpolasi.

BAB III IMPLEMENTASI

A. Implementasi Matriks

Class Matrix

- Atribut

Nama	Tipe	Deskripsi
ROW_CAP	Final int	Baris maksimal yang dapat dimiliki matriks
COL_CAP	Final int	Kolom maksimal yang dapat dimiliki matriks
memory	float[][]	Buffer yang menyimpan nilai matriks
rowEff	int	Jumlah baris
colEff	int	Jumlah kolom

- Method

Nama	Tipe	Parameter	Deskripsi
Matrix	public Constructor	int rowEff, int colEff	Membuat matriks kosong dengan dimensi rowEff x colEff
isMatrixIdxValid	public static boolean	int i , int j	Mengirimkan true jika i, j adalah index yang valid untuk matriks apa pun
getLastIdxRow	public static int	Matrix m	Mengirimkan Index baris terbesar m
getLastIdxCol	public static int	Matrix m	Mengirimkan Index kolom terbesar m
isIdxEff	public static boolean	Matrix m, int i , int j	Mengirimkan true jika i,j adalah Index efektif bagi m
getElmtDiagonal	public static float	Matrix m, int i	Mengirimkan elemen m(i,i)
copyMatrix	public static void	Matrix mIN, Matrix mOut	Melakukan assignment mOut <-

			mIN
readMatrix	public static void	Matrix m, int nRow, int nCol	Mengisi element matrix m dengan inputan user
displayMatrix	public static void	Matrix m	Menampilkan matriks dengan format matriks pada umumnya, contoh : 1 2 3 4 5 6 7 8 9
addMatrix	public static Matrix	Matrix m, Matrix n	Mengirim hasil penjumlahan m1 + m2
subtractMatrix	public static Matrix	Matrix m1, Matrix m2	Mengirim hasil m1 - m2
multiplyMatrix	public static Matrix	Matrix m1, Matrix m2	Mengirim hasil m1*m2
multiplyMarixWithMod	public static Matrix	Matrix m1, Matrix m2, int mod	Mengirim hasil perkalian matriks: salinan (m1 * m2)%mod, artinya setiap elemen matrix hasil perkalian m1*m2 dilakukan modulo terhadap mod
multiplyByConst	public static Matrix	Matrix m, float x	mengirim hasil perkalian setiap elemen m dengan x
pMultiplyByConst	public static void	Matrix m, float x	Mengalikan setiap elemen m dengan k
isMatrixEqual	public static boolean	Matrix m1, Matrix m2	Mengirimkan true jika m1 = m2, yaitu count(m1) = count(m2) dan untuk setiap i,j yang merupakan index baris dan kolom m1(i,j) = m2(i,j)
isMatrixNotEqual	public static boolean	Matrix m1, Matrix m2	Mengirimkan negasi

			dari isMatrixEqual
isMatrixSizeEqual	public static boolean	Matrix m, Matrix n	Mengirimkan true jika ukuran efektif matriks m1 sama dengan ukuran efektif m2.
countElmt	public static int	Matrix m	Mengirimkan banyak elemen
isSquare	public static boolean	Matrix m	Mengirimkan true jika m adalah matriks dengan ukuran baris dan kolom sama
isSymetric	public static boolean	Matrix m	Mengirimkan true jika m adalah matriks simetri
isIdentity	public static boolean	Matrix m	Mengirimkan true jika m adalah matriks satuan
transpose	public static Matrix	Matrix m	Mengembalikan salinan transpose dari m
pTranspose	public static void	Matrix m	Melakukan operasi transpose pada m
Swap	public static void	Matrix m, int Row1, int Row2, int nSwap	Menukar Row1 dan Row2 serta menambahkan nSwap dengan 1
MultiplyRow	public static void	Matrix m, int Row, float multiplier	Mengalikan semua elemen pada di baris Row pada Matrix m dengan multiplier
AddRowByRow	public static void	Matrix m, int Row1, int Row2, float x	Menambahkan baris row1 dengan baris row2*x
coefficientMatrix	public static Matrix	Matrix m	Mengembalikan koefisien dari matriks augmented m

B. Implementasi Input/Output

Class ReadFile

- Method

Nama	Tipe	Parameter	Deskripsi
countLine	public static int	String namaFile	Mengembalikan banyak baris pada file namaFile
isFileExist	public static boolean	String namaFile	Mengirimkan true jika namaFile ada di folder test
parseFile	public static Matrix	Matrix m, String nama_file	Mengirimkan matriks dari nama_file
readMatrixFromFile	public static Matrix	String namaFile	Mengirimkan matriks dari namaFile (khusus kasus interpolasi, regresi, dan bikubik)
readBottomLine	public static float[]	String namaFile	Mengirimkan baris terbawah dari namaFile (khusus kasus interpolasi, regresi, dan bikubik)

Class WriteToFile

- Method

Nama	Tipe	Parameter	Deskripsi
MatrixtoString	public static String	Matrix m	Mengembalikan matriks dalam bentuk string
ArrayOfStringtoString	public static String	float[] lis	Mengubah nilai nilai di dalam float menjadi string
writeFile	public static void	String x, String fileName	Membuat file .txt dengan nama fileName yang berisi x

C. Implementasi Eliminasi Gauss

Class Gauss

- Method

Nama	Tipe	Parameter	Deskripsi
countZero	public static int	int x, Matrix m	Mengirimkan banyaknya angka 0 pada baris x.
zeroDown	public static void	Matrix m	Melakukan swapping pada m sehingga baris dengan 0 awal terbanyak ada di bawah.
firstNonZeroIdx	public static int	Matrix m, int row	Mengembalikan index element bukan 0 pertama pada row
uniqueSolution	public static boolean	Matrix m	Menerima matrix m yang merupakan matrix echelon dan mengembalikan true bila m memiliki solusi unik.
infiniteSolution	public static boolean	Matrix m	Menerima matrix m yang merupakan matrix echelon dan mengembalikan true bila m memiliki solusi banyak.
noSolution	public static boolean	Matrix m	Menerima matrix m yang merupakan matrix echelon dan mengembalikan true bila m tidak memiliki solusi.
containsString	public static boolean	String[] stringArray, String targetString	Mengembalikan true bila targetString terdapat pada stringArray
makeSubMatrix	public static Matrix	Matrix m, int parameterAmmount	Mengembalikan matrix m yang barisnya dihapus sebanyak parameterAmmount dari bawah dan kolomnya dihapus sebanyak parameterAmmount

			dari kanan.
makeEchelon	public static void	Matrix m	Membuat matriks m menjadi matriks echelon
printArrayNeatly	public static void	String[] arr	Mencetak array of string
backwardSubstitution	public static float[]	Matrix m	Menerima matriks echelon m dan mengembalikan solusi dari matriks tersebut dalam bentuk array of float (hanya berlaku untuk unique solution)
findIndex	public static int	float[] arr, float target	Mengembalikan index bila target ada pada arr, mengembalikan -1 bila tidak ada.
safeStringToFloat	public static float	String str	Mengubah string menjadi float bila memungkinkan.
gauss	public static float[]	Matrix m	Melakukan eliminasi gauss pada m dan memberikan hasilnya dalam bentuk array of float
gaussString	public static String	Matrix m	Serupa dengan Gauss tapi mengembalikan hasilnya dalam bentuk string.

D. Implementasi Eliminasi Gauss-Jordan

Class Gauss Jordan

- Method

Nama	Tipe	Parameter	Deskripsi
isHomogen	public static boolean	Matrix m	Mengembalikan true bila sebuah matriks adalah matriks homogen.
echelontoEchelonBaris	public static void	Matrix m	Mengubah matriks echelon menjadi matriks echelon baris
SPLGaussJordan	public static float[]	Matrix m	Menerima matriks augmented m dan mengembalikan solusinya dalam bentuk float[] bila unik dengan menggunakan metode eliminasi Gauss-Jordan
string SPLGaussJordan	public static String	Matrix m	Mirip dengan SPLGaussJordan tapi akan mengembalikan string hasilnya, pada kondisi apa pun.

E. Implementasi Reduksi Baris

Class reduksiBaris

- Method

Nama	Tipe	Parameter	Deskripsi
getDeterminant	public static float	Matrix m	mengembalikan determinan dari matriks m dengan metode reduksi baris.

F. Implementasi Matriks Balikan

Class Invers

- Method

Nama	Tipe	Parameter	Deskripsi
------	------	-----------	-----------

MakelIdentity	public static Matrix	Int Row, int Col	Menghasilkan matriks identitas ukuran Row x Col
MatrixSoal	public static Matrix	Matrix m	Mengeluarkan matriks m yang dipotong kolom terakhirnya
InverseWithGaussJordan	public static Matrix	Matrix m	Mengirimkan hasil balikan m dengan metode matriks identitas dan OBE
InverseWithCofactor	public static Matrix	Matrix m	Mengirimkan hasil balikan m dengan ekspansi adjoin
SolusiSPLDenganInvers	public static float[]	Matrix m, int pilihan	Mengirimkan koefisien SPL dengan metode yang dipilih

G. Implementasi Kofaktor

Class Cofactor

- Method

Nama	Tipe	Parameter	Deskripsi
cofactorDeterminant	public static float	Matrix m	Menghitung determinan matriks m dengan menggunakan metode kofaktor
entryCofactor	public static float	Matrix m, int idxRow, int idxCol	Menghasilkan kofaktor untuk entri dengan indeks idxRow, idxCol
cofactorMatrix	public static Matrix	Matrix m	Mencari matriks kofaktor, yaitu matriks yang tiap elemennya adalah kofaktor dari tiap entri

H. Implementasi Kaidah Cramer

Class Cramer

- Method

Nama	Tipe	Parameter	Deskripsi
replaceColumnWithConstant	public static Matrix	Matrix m, int idxCol	Mengganti kolom tertentu dari matriks koefisien dengan kolom terakhir matriks augmented
SPLCramer	public static float[]	Matrix m	Mencari solusi SPL dengan kaidah Cramer dan mengembalikan inlainya

I. Implementasi Interpolasi Polinomial

Class Interpolasi_Polinomial

- Atribut

Nama	Tipe	Deskripsi
scanner	Scanner	Scanner yang digunakan untuk menerima input dari user.

- Method

Nama	Tipe	Parameter	Deskripsi
titikInput	public static Matrix		Meminta dan mengembalikan titik - titik yang digunakan dalam melakukan interpolasi
interpolasiPolinomial	public static float		Fungsi yang akan melakukan interpolasi polinomial terhadap titik - titik yang diinput, menampilkan persamaannya, lalu mengembalikan hasil bila suatu x dimasukkan ke dalam persamaan tersebut.

J. Implementasi Regresi Linear Berganda

Class Regresi

- Atribut

Nama	Tipe	Deskripsi
koef	float[]	Buffer yang menyimpan nilai koefisien hasil regresi
nilai	float	Hasil perhitungan dari variabel dikalikan dengan koefisien

- Method

Nama	Tipe	Parameter	Deskripsi
readPoint	public static Matrix	int n, int m	Membaca matriks berukuran $m \times n + 1$
sumCol	public static float	Matrix m, int IdxCol	Menjumlahkan total nilai kolom ke-IdxCol dari m
sumColTimesCol	public static float	Matrix m, int IdxCol1, int IdxCol2	Menjumlahkan total nilai kolom ke-IdxCol1 dikali kolom ke IdxCol2 dari m
normalEstimationEquation	public static Matrix	Matrix m	Menghasilkan matriks persamaan linear dari m
prosesRegresiBerganda	public static Matrix	int n, int m	Memberikan matriks persamaan linear dari jumlah peubah (n) dan jumlah data (m)
ambilHasil	public static float[]	Matrix hasil	Menyimpan solusi SPL hasil ke dalam suatu array of float
hitungNilai	public static float	float[] solusi, float[] peubah, int n	Mengembalikan nilai koefisien dikali pengubah
tampilHitungNilai	public static String	float result	Membuat "f(xk) = result"
tampilHasil	public static String	float[] solusi	Membuat string "f(x)

			$= a_0 + a_1x_1 + \dots + a_nx_n$
regresiBerganda	public static Regresi	int n, float[] koefisien, float[] peubah	Memasukan nilai koefisien dan result ke dalam Regresi

K. Implementasi Interpolasi *Bicubic-Spline*

Class BicubicSpline

- Method

Nama	Tipe	Parameter	Deskripsi
interpolation	public static float	Matrix m, float targetX, float targetY	Menghitung hasil aproksimasi fungsi oleh interpolasi <i>Bicubic-Spline</i>

BAB IV EKSPERIMEN

1. Mencari Solusi SPL $AX = b$

Soal	$A = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 2 & 5 & -7 & -5 \\ 2 & -1 & 1 & 3 \\ 5 & 2 & -4 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ -2 \\ 4 \\ 6 \end{bmatrix}$
Metode Gauss	TIDAK ADA SOLUSI
Metode Gauss-Jordan	TIDAK ADA SOLUSI
Metode Matriks Balikan	Solusi SPL tidak dapat ditemukan dengan metode invers.
Metode Cramer	Solusi SPL tidak dapat ditemukan dengan metode Cramer.

Soal	$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 0 & -3 & 0 \\ 2 & -1 & 0 & 1 & -1 \\ -1 & 2 & 0 & -2 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 6 \\ 5 \\ -1 \end{bmatrix}$
Metode Gauss	<p>BANYAK SOLUSI</p> <p>2</p> <p>P1, P2, null, null, (3.0-1P1)/-1.0</p> <p>x1=0.0</p>
Metode Gauss-Jordan	<p>BANYAK SOLUSI</p> <p>2</p> <p>P1, P2, null, null, (3.0-1P1)/-1.0</p> <p>x1=0.0</p>

Metode Matriks Balikan	Solusi SPL tidak dapat ditemukan dengan metode invers.
Metode Cramer	Solusi SPL tidak dapat ditemukan dengan metode Cramer.

Soal	$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}$
Metode Gauss	$x_1 = 0.0$
Metode Gauss-Jordan	$x_1 = 0.0$
Metode Matriks Balikan	Solusi SPL tidak dapat ditemukan dengan metode invers.
Metode Cramer	Solusi SPL tidak dapat ditemukan dengan metode Cramer.

n = 6

Soal	$H = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \dots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \dots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \dots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \dots & \frac{1}{2n+1} \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$
Metode Gauss	$x_1 = -1.9175872E7, x_2 = 3.835388E8, x_3 = -1.7259727E9, x_4 = 2.6848932E9, x_5 = -1.3424632E9, x_6 = 629.7805$

Metode Gauss-Jordan	$x_1=2.344368E7, x_2=-4.155682E8, x_3=1.4704228E9, x_4=-1.04420736E9, x_5=-1.3424632E9, x_6=629.7805$
Metode Matriks Balikan	$x_1=7.9273067, x_2=-32.74581, x_3=50.268673, x_4=-30.411867, x_5=-45.49167, x_6=56.34404$
Metode Cramer	$x_1=7.9273067, x_2=-32.74581, x_3=50.268673, x_4=-30.411867, x_5=-45.49167, x_6=56.34404$

2. SPL berbentuk matriks augmented

Soal	$\begin{bmatrix} 1 & -1 & 2 & -1 & -1 \\ 2 & 1 & -2 & -2 & -2 \\ -1 & 2 & -4 & 1 & 1 \\ 3 & 0 & 0 & -3 & -3 \end{bmatrix}.$
Metode Gauss	<p>BANYAK SOLUSI</p> <p>2</p> <p>P1, P2, $(-1P2)/-2.0$, $(-1.0-1P1)/-1.0$</p>
Metode Gauss-Jordan	<p>BANYAK SOLUSI</p> <p>2</p> <p>P1, P2, $(-1P2)/-2.0$, $(-1.0-1P1)/-1.0$</p>
Metode Matriks Balikan	<p>Solusi SPL tidak dapat ditemukan dengan metode invers.</p>
Metode Cramer	<p>Solusi SPL tidak dapat ditemukan dengan metode Cramer.</p>

Soal	$\begin{bmatrix} 2 & 0 & 8 & 0 & 8 \\ 0 & 1 & 0 & 4 & 6 \\ -4 & 0 & 6 & 0 & 6 \\ 0 & -2 & 0 & 3 & -1 \\ 2 & 0 & -4 & 0 & -4 \\ 0 & 1 & 0 & -2 & 0 \end{bmatrix}.$
Metode Gauss	$x_1=0.0, x_2=2.0, x_3=1.0, x_4=1.0$
Metode Gauss-Jordan	$x_1=0.0, x_2=2.0, x_3=1.0, x_4=1.0$
Metode Matriks Balikan	$x_1=0.0, x_2=2.0, x_3=1.0, x_4=1.0$
Metode Cramer	$x_1=0.0, x_2=2.0, x_3=1.0, x_4=1.0$

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

Metode Matriks Balikan	$x_1=5.1200104, x_2=-128.0, x_3=48.0, x_4=96.0, x_5=-282.8, x_6=80.0, x_7=96.0, x_8=144.0, x_9=-65.46$
Metode Cramer	$x_1=470.25824, x_2=28096.975, x_3=4.1251593, x_4=83663.125, x_5=-4248.8633, x_6=-17.06528, x_7=-13.000001, x_8=1.0, x_9=1.0$

Soal	$\begin{aligned} \text{A: } m_{A_{in}} + Q_{BA}x_B - Q_{AB}x_A - Q_{AC}x_A &= 0 \\ \text{B: } Q_{AB}x_A - Q_{BA}x_B - Q_{BC}x_B &= 0 \\ \text{C: } m_{C_{in}} + Q_{AC}x_A + Q_{BC}x_B - Q_{C_{out}}x_C &= 0 \end{aligned}$ <p>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 $Q_{C_{out}} = 150 \text{ m}^3/\text{s}$ dan $m_{A_{in}} = 1300$ dan $m_{C_{in}} = 200 \text{ mg/s}$.</p>
Metode Gauss	$x_1=14.444446, x_2=7.2222233, x_3=10.000001$
Metode Gauss-Jordan	$x_1=14.444446, x_2=7.2222233, x_3=10.000001$
Metode Matriks Balikan	$x_1=-1300.0, x_2=0.0, x_3=-200.0$
Metode Cramer	$x_1=14.444445, x_2=7.222223, x_3=10.000001$

Studi Kasus Interpolasi

Soal

- a. Gunakan tabel di bawah ini untuk mencari polinom interpolasi dari pasangan titik-titik yang terdapat dalam tabel. Program menerima masukan nilai x yang akan dicari nilai fungsi $f(x)$.

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

Lakukan pengujian pada nilai-nilai berikut:

$x = 0.2$	$f(x) = ?$
$x = 0.55$	$f(x) = ?$
$x = 0.85$	$f(x) = ?$
$x = 1.28$	$f(x) = ?$

x = 0.2

```
Persamaannya adalah:  
P(x)= -0.4459818 + 7.800377X + -42.326904X^2 + 102.95352X^3 + -111.879906X^4 + 44.762352X^5 + -2.2676948E-6X^6  
Jawabannya adalah: 0.07996157
```

x = 0.55

```
Persamaannya adalah:  
P(x)= -0.4459818 + 7.800377X + -42.326904X^2 + 102.95352X^3 + -111.879906X^4 + 44.762352X^5 + -2.2676948E-6X^6  
Jawabannya adalah: 0.18433708
```

x =0.85

```
Persamaannya adalah:  
P(x)= -0.4459818 + 7.800377X + -42.326904X^2 + 102.95352X^3 + -111.879906X^4 + 44.762352X^5 + -2.2676948E-6X^6  
Jawabannya adalah: 0.28876218
```

x= 1.28

```
Persamaannya adalah:  
P(x)= -0.4459818 + 7.800377X + -42.326904X^2 + 102.95352X^3 + -111.879906X^4 + 44.762352X^5 + -2.2676948E-6X^6  
Jawabannya adalah: 9.576193
```

Soal

- b. Jumlah kasus positif baru Covid-19 di Indonesia semakin fluktuatif dari hari ke hari. Di bawah ini diperlihatkan jumlah kasus baru Covid-19 di Indonesia mulai dari tanggal 17 Juni 2022 hingga 31 Agustus 2022:

Tanggal	Tanggal (desimal)	Jumlah Kasus Baru
17/06/2022	6,567	12.624
30/06/2022	7	21.807
08/07/2022	7,258	38.391
14/07/2022	7,451	54.517
17/07/2022	7,548	51.952
26/07/2022	7,839	28.228
05/08/2022	8,161	35.764
15/08/2022	8,484	20.813
22/08/2022	8,709	12.408
31/08/2022	9	10.534

untuk semua test case:

```
Persamaannya adalah:  
P(x)= Infinity + -2.3569127E38X + 9.252241E31X^2 + -6.910396E29X^3 + 2.3640621E27X^4 + -4.5649633E24X^5 + -1.5954525E16X^6 + 1.0678929E12X^7 + -3.3713984E7X^8 + 5749.1704X^9  
Jawabannya adalah: NaN
```

soal :

Sederhanakan fungsi $f(x)$ yang memenuhi kondisi

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

dengan polinom interpolasi derajat n di dalam selang $[0, 2]$.

Sebagai contoh, jika $n = 5$, maka titik-titik x yang diambil di dalam selang $[0, 2]$ berjarak $h = (2 - 0)/5 = 0.4$.

Studi Kasus Regresi Linear Berganda

Soal	<div>Table 12.1: Data for Example 12.1</div> <table><tr><th>Nitrous Oxide, y</th><th>Humidity, x_1</th><th>Temp., x_2</th><th>Pressure, x_3</th><th>Nitrous Oxide, y</th><th>Humidity, x_1</th><th>Temp., x_2</th><th>Pressure, x_3</th></tr><tr><td>0.90</td><td>72.4</td><td>76.3</td><td>29.18</td><td>1.07</td><td>23.2</td><td>76.8</td><td>29.38</td></tr><tr><td>0.91</td><td>41.6</td><td>70.3</td><td>29.35</td><td>0.94</td><td>47.4</td><td>86.6</td><td>29.35</td></tr><tr><td>0.96</td><td>34.3</td><td>77.1</td><td>29.24</td><td>1.10</td><td>31.5</td><td>76.9</td><td>29.63</td></tr><tr><td>0.89</td><td>35.1</td><td>68.0</td><td>29.27</td><td>1.10</td><td>10.6</td><td>86.3</td><td>29.56</td></tr><tr><td>1.00</td><td>10.7</td><td>79.0</td><td>29.78</td><td>1.10</td><td>11.2</td><td>86.0</td><td>29.48</td></tr><tr><td>1.10</td><td>12.9</td><td>67.4</td><td>29.39</td><td>0.91</td><td>73.3</td><td>76.3</td><td>29.40</td></tr><tr><td>1.15</td><td>8.3</td><td>66.8</td><td>29.69</td><td>0.87</td><td>75.4</td><td>77.9</td><td>29.28</td></tr><tr><td>1.03</td><td>20.1</td><td>76.9</td><td>29.48</td><td>0.78</td><td>96.6</td><td>78.7</td><td>29.29</td></tr><tr><td>0.77</td><td>72.2</td><td>77.7</td><td>29.09</td><td>0.82</td><td>107.4</td><td>86.8</td><td>29.03</td></tr><tr><td>1.07</td><td>24.0</td><td>67.7</td><td>29.60</td><td>0.95</td><td>54.9</td><td>70.9</td><td>29.37</td></tr></table> <div>Source: Charles T. Hare, "Light-Duty Diesel Emission Correction Factors for Ambient Conditions," EPA-600/2-77-116. U.S. Environmental Protection Agency.</div> <p>Gunakan <i>Normal Estimation Equation for Multiple Linear Regression</i> untuk mendapatkan regresi linear berganda dari data pada tabel di atas, kemudian estimasi nilai Nitrous Oxide apabila Humidity bernilai 50%, temperatur 76°F, dan tekanan udara sebesar 29.30.</p>	Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3	Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3	0.90	72.4	76.3	29.18	1.07	23.2	76.8	29.38	0.91	41.6	70.3	29.35	0.94	47.4	86.6	29.35	0.96	34.3	77.1	29.24	1.10	31.5	76.9	29.63	0.89	35.1	68.0	29.27	1.10	10.6	86.3	29.56	1.00	10.7	79.0	29.78	1.10	11.2	86.0	29.48	1.10	12.9	67.4	29.39	0.91	73.3	76.3	29.40	1.15	8.3	66.8	29.69	0.87	75.4	77.9	29.28	1.03	20.1	76.9	29.48	0.78	96.6	78.7	29.29	0.77	72.2	77.7	29.09	0.82	107.4	86.8	29.03	1.07	24.0	67.7	29.60	0.95	54.9	70.9	29.37
Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3	Nitrous Oxide, y	Humidity, x_1	Temp., x_2	Pressure, x_3																																																																																		
0.90	72.4	76.3	29.18	1.07	23.2	76.8	29.38																																																																																		
0.91	41.6	70.3	29.35	0.94	47.4	86.6	29.35																																																																																		
0.96	34.3	77.1	29.24	1.10	31.5	76.9	29.63																																																																																		
0.89	35.1	68.0	29.27	1.10	10.6	86.3	29.56																																																																																		
1.00	10.7	79.0	29.78	1.10	11.2	86.0	29.48																																																																																		
1.10	12.9	67.4	29.39	0.91	73.3	76.3	29.40																																																																																		
1.15	8.3	66.8	29.69	0.87	75.4	77.9	29.28																																																																																		
1.03	20.1	76.9	29.48	0.78	96.6	78.7	29.29																																																																																		
0.77	72.2	77.7	29.09	0.82	107.4	86.8	29.03																																																																																		
1.07	24.0	67.7	29.60	0.95	54.9	70.9	29.37																																																																																		
Metode Gauss	<div>$f(x) = -3.5119526 + x_1(-0.0026243355) + x_2(7.99263E-4) + x_3(0.15429525)$$f(x_k) = 0.9384253$</div>																																																																																								

Studi Kasus Interpolasi *Bicubic-Spline*

Soal	$\begin{pmatrix} 21 & 98 & 125 & 153 \\ 51 & 101 & 161 & 59 \\ 0 & 42 & 72 & 210 \\ 16 & 12 & 81 & 96 \end{pmatrix}$ <p>Tentukan nilai:</p> <p>$f(0, 0) = ?$ $f(0.5, 0.5) = ?$ $f(0.25, 0.75) = ?$ $f(0.1, 0.9) = ?$</p>

Jawaban	$f(0.0, 0.0) = 21.0$ $f(0.5, 0.5) = 124.70367$ $f(0.25, 0.75) = 81.16504$ $f(0.1, 0.9) = 97.97921$
---------	--

Link to Library:

https://drive.google.com/file/d/1WVJcSmNKcc7ltr7T9NBuxbvcJV_OawOV/view?usp=drive_link

Sources:

1. Cue Maths <https://www.cuemath.com/algebra/triangular-matrix/>
2. Britannica, T. Editors of Encyclopaedia (2023, September 21). Cramer's rule. Encyclopedia Britannica. <https://www.britannica.com/science/Cramers-rule>
3. Rowe, Daniel. https://www.mssc.mu.edu/~daniel/pubs/RoweTalkMSCS_BiCubic.pdf
4. Munir, Rinaldi. <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/Algeo-04-Tiga-Kemungkinan-Solusi-SPL-2023.pdf>
5. Munir, Rinaldi. <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/Algeo-05-Sistem-Persamaan-Linier-2023.pdf>
6. Munir, Rinaldi <https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2023-2024/Algeo-03-Sistem-Persamaan-Linier-2023.pdf>