**TUGAS BESAR 1**

**IF2123 ALJABAR LINIER DAN GEOMETRI**

**SISTEM PERSAMAAN LINIER, DETERMINAN, DAN APLIKASINYA**

**SEMESTER 1 TAHUN 2020/2021**

oleh

Rayhan Alghifari Fauzta / 13519039

Irvin Andryan Pratomo / 13519162

Reyhan Emyr Arrosyid / 13519167

Two people standing in front of a television

Description automatically generated

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA  
INSTITUT TEKNOLOGI BANDUNG**

**BANDUNG**

**2020**

**BAB I**

**DESKRIPSI MASALAH**

Tugas besar ini merupakan salah satu bentuk pengaplikasian konsep sistem persamaan linier dan determinan yang telah dipelajari di kuliah. Pada tugas ini, mahasiswa diminta untuk membuat sebuah program yang dapat menerima input berbentuk matriks serta melakukan manipulasi dan operasi terhadap matriks tersebut.

Berikut gambaran spesifikasi program:

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

3 4.5 2.8 10 12

-3 7 8.3 11 -4

0.5 -10 -9 12 0

1. Untuk persoalan menghitung determinan dan matriks balikan, masukan dari *keyboard* adalah *n* dan koefisien *aij*. Masukan dari *file* berbentuk matriks, setiap elemen matriks dipisah oleh spasi. Misalnya,

3 4.5 2.8 10

-3 7 8.3 11

0.5 -10 -9 12

1. Untuk persoalan interpolasi, masukannya jika dari *keyboard* adalah *n*, (*x*0, *y*0), (*x*1, *y*1), ..., (*xn*, *yn*), dan nilai *x* yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung. Misalnya jika titik-titik datanya adalah (8.0, 2.0794), (9.0, 2.1972), dan (9.5, 2.2513), maka di dalam file text ditulis sebagai berikut:

8.0 2.0794

9.0 2.1972

9.5 2.2513

1. Untuk persoalan regresi, masukannya jika dari *keyboard* adalah *n* (jumlah peubah *x*), semua nilai-nilai *x*1*i*, *x*2*i*, ..., *xni*, nilai *yi*, dan nilai-nilai *xk* yang akan ditaksir nilai fungsinya. Jika masukannya dari file, maka titik-titik dinyatakan pada setiap baris tanpa koma dan tanda kurung.
2. Untuk persoalan SPL, luaran (*output*) program adalah solusi SPL. Jika solusinya tunggal, tuliskan nilainya. Jika solusinya tidak ada, tuliskan solusi tidak ada, jika solusinya banyak, maka tuliskan solusinya dalam bentuk parametrik (misalnya *x*4 = -2, *x*3 = 2*s* – *t*, *x*2 = *s*, dan *x*1 = *t*.)
3. Untuk persoalan determinan dan matriks balikan, maka luarannya sesuai dengan persoalan masing-masing
4. Untuk persoalan polinom interpolasi dan regresi, luarannya adalah persamaan polinom/regresi dan taksiran nilai fungsi pada *x* yang diberikan.
5. Luaran program harus dapat ditampilkan **pada layar komputer dan dapat disimpan ke dalam file**.
6. Bahasa program yang digunakan adalah Java.
7. Program **tidak harus** berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas *Eclipse* misalnya).
8. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

MENU

1. Sistem Persamaaan Linier

2. Determinan

3. Matriks balikan

4. Interpolasi Polinom

5. Regresi linier berganda

6. Keluar

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

1. Metode eliminasi Gauss

2. Metode eliminasi Gauss-Jordan

3. Metode matriks balikan

4. Kaidah Cramer

Begitu juga untuk pilihan menu nomor 2 dan 3.

**BAB II**

**TEORI SINGKAT**

**Eliminasi Gauss dan Gauss Jordan**

Eliminasi Gauss yang ditemukan oleh Carl Friedrich Gauss adalah salah satu metode untuk mencari solusi sistem persamaan linear dengan memanfaatkan matriks. Sistem persamaan linear tersebut diimplementasikan ke dalam matriks augmented, seperti contoh berikut.

A picture containing icon

Description automatically generated

Dalam eliminasi Gauss kemudian dilakukan operasi baris elementer (OBE) terhadap matriks augmented tersebut sehingga diperoleh matriks eselon baris, yaitu matriks segitiga atas dengan elemen Mij = 1, saat i = j.

A picture containing diagram

Description automatically generated

Dari matriks eselon baris tersebut diperoleh solusi SPL sebagai berikut.

Text, letter

Description automatically generated

Eliminasi Gauss-Jordan adalah lanjutan dari metode eliminasi Gauss yang dikembangkan oleh Wilhelm Jordan. Metode ini memanfaatkan bentuk matriks eselon baris dari eliminasi Gauss dan melakukan OBE lebih lanjut sehingga diperoleh bentuk matriks eselon baris tereduksi

A picture containing diagram

Description automatically generated

Dari matriks eselon baris tereduksi tersebut, diperoleh X1 = 1, X2 = 2, dan X3 = 3.

Bentuk matriks eselon baris yang dihasilkan dari eliminasi Gauss atau Gauss-Jordan, dapat digunakan untuk menentukan kemungkinan solusi dari sistem persamaan linear yang ada. Bentuk-bentuk tersebut adalah sebagai berikut.

A picture containing diagram

Description automatically generated

**Matriks Balikan**

Matriks balikan atau invers adalah teorema matriks yang menyatakan bahwa jika suatu matriks A memiliki balikan berupa matriks B sedemikian sehingga

AB = BA = I

Matriks invers dapat disimbolkan dengan tanda A-1, sehingga sifat matriksnya menjadi

AA-1 = A-1A = I

Untuk suatu matriks 2x2 balikannya dapat diperoleh dengan cara berikut

A picture containing diagram

Description automatically generated

ad – bc dalam perhitungan tersebut adalah determinan dari matriks A, sehingga apabila sebuah matriks memiliki determinan 0, maka matriks tersebut tidak memiliki balikan. Untuk matriks berukuran nxn, balikannya dapat dihitung dengan menggunakan memanfaatkan metode Gauss-Jordan.

A close up of a screen

Description automatically generated

Dari bentuk tersebut, dilakukan OBE sehingga diperoleh :

A picture containing text

Description automatically generated

Maka, balikan dari matriks A adalah :

A picture containing object, clock, orange, people

Description automatically generated

**Determinan**

Misalkan A adalah sebuah matriks berukuran nxn,

Table

Description automatically generated

Maka, determinan dari matriks A dapat dilambangkan sebagai

A picture containing text

Description automatically generated

Determinan dari sebuah matriks nxn dapat dicari dengan memanfaatkan matriks segitiga (upper triangular atau lower triangular).

A picture containing graphical user interface

Description automatically generated

Hal-hal yang menjadi teorema dari determinan matriks adalah sebagai berikut:

1. Jika A mengandung baris / kolom yang semua anggotanya nol, maka det(A) = 0
2. Jika AT adalah transpose matriks A, maka det(A) = det(AT)
3. Jika A = BC, maka det(A) = det(B)det(C)
4. Matriks memiliki balikan jika det(A) ≠ 0
5. det(A-1) = 1/det(A)

**Matriks Kofaktor dan Adjoint Matriks**

Misalkan A adalah matriks nxn dan Cij adalah kofaktor entri aij, maka kofaktor matriks A adalah

**A picture containing table

Description automatically generated**

Adjoint dari matriks A adalah transpose dari matriks kofaktor A. Adjoin matriks dapat dimanfaatkan untuk mencari matriks balikan, menggunakan persamaan sebagai berikut

Chart

Description automatically generated

**Kaidah Cramer**

Jika Ax = b adalah SPL yang terdiri dari n persamaan linier dan n variabel / peubah, sedemikian sehingga det(A) ≠ 0, maka SPL tersebut memiliki solusi unik, yaitu

Text

Description automatically generated

Aj adalah matriks yang diperoleh dengan mengganti entri pada kolom ke-j matriks A dengan

A picture containing diagram

Description automatically generated

**Interpolasi Polinom**

Pada interpolasi polinom, diberikan sebuah n+1 buah titik (X0, Y0), …., (Xn, Yn) sedemikian sehingga terdapat Pn(Xi) yang menginterpolasikan himpunan titik-titik {(X0, Y0), …., (Xn, Yn)}. Pn(Xi) dapat digunakan untuk memprediksi nilai Y dari suatu nilai X pada selang [X0…Xn]. Dalam persamaan Lagrange, polynomial p(x) dapat diperoleh sebagai berikut



Dengan menggunakan matriks, interpolasi polinomial dapat dituliskan sebagai berikut

A close up of a clock

Description automatically generated

**Regresi Linier Berganda**

Regresi linier berganda digunakan untuk menjelaskan hubungan suatu variabel tak bebas (Y) dengan dua atau lebih variabel bebas (X1, X2, X3, …,Xn). Regresi linier berganda bertujuan untuk memprediksi nilai dari variabel tak bebas (Y), jika nilai-nilai variabel bebasnya (X) diketahui. Secara matematik, regresi linier berganda dapat dituliskan sebagai berikut

Y = a + b1 X1i + b2X2i + … + bn Xni

Regresi linier berganda juga dapat diselesaikan dengan menggunakan matriks. Misalkan suatu persamaan regresi sebagai berikut

Text, letter

Description automatically generated

Sebagai contoh, persamaan tersebut dapat dituliskan seperti ini :

Text, letter

Description automatically generated

Dalam tugas ini, bentuk persamaan tersebut dapat dibuat matriks yang digunakan untuk melakukan penyelesaian dengan metode eliminasi Gauss.

**BAB III**

**IMPLEMENTASI PROGRAM**

**3.1. Class Matriks**

**3.1.1. Atribut**

|  |
| --- |
| int M  {Jumlah baris dari matriks} |
| int N  {Jumlah baris dari matriks} |
| double[][] Mat  {Nilai-nilai matriks dalam array dua dimensi} |

**3.1.2. Method**

|  |
| --- |
| Matriks (int M, int N)  { Konstruktor Matriks, membentuk matriks kosong dengan ukuran MxN } |
| Matriks (File f)  { Membaca matriks dari sebuah file } |
| Matriks (double [ ][ ] A)  { Membentuk matriks dari array 2 dimensi } |
| Matriks clone ()  { Membentuk matriks baru hasil copy matriks input } |
| Void inputMatriks ()  { I.S. M dan N terdefinisi }  { F.S. Terbentuk Matriks berukuran M x N dengan input dari pengguna } |
| Void printMatriks ()  { I.S. Matriks terdefinisi }  { F.S. Matriks ditampilkan di layar } |
| Void swapRow (int i, int j)  { I.S. Matriks terdefinisi }  { F.S. Matriks memiliki baris yang sudah ditukar posisinya } |
| int leadingCoef (int i)  { Mencari dan mengembalikan indeks kolom leading coefficient di baris i } |
| boolean rowZero (int i)  { Melakukan cek apakah satu baris bernilai nol semua atau tidak } |
| Void gauss ()  { I.S. Matriks terdefinisi }  { F.S. Terbentuk matriks eselon baris } |
| Void gaussJordan ()  { I.S. Matriks terdefinisi }  { F.S. Terbentuk matriks eselon baris tereduksi } |
| Matriks transpose ()  { Mengembalikan matriks transpose } |
| Matriks getCofactorMatriks ()  { Mengembalikan matriks kofaktor } |
| double detRowReduction ()  { Menghitung determinan matriks menggunakan metode reduksi baris, dan mengembalikan nilainya (prekondisi : matriks persegi) } |
| double detCofactor ()  { Menghitung determinan matriks menggunakan metode kofaktor dan mengembalikan nilainya (prekondisi : matriks persegi) } |
| Matriks inverseCofactor ()  { Mengembalikan matriks balikan dengan metode ekspansi kofaktor (prekondisi: matriks persegi dan determinan tidak 0) } |
| Matriks inverseGaussJordan  { Mengembalikan matriks balikan dengan metode gauss jordan (prekondisi: matriks persegi dan determinan tidak 0) } |

**3.2. Class InverseSolver**

**3.2.1. Atribut**

|  |
| --- |
| Matriks mat  {Matriks yang akan dicari inversenya} |
| Matriks inv  {Matriks inverse yang didapat dari mat} |
| boolean toFile  {Nilai boolean untuk menandakan output program ke file atau layar} |

**3.2.2. Method**

|  |
| --- |
| InverseSolver ()  { Konstruktor yang menerima input matriks dari file atau dari keyboard pengguna } |
| Void printInv ()  { I.S. Inverse matriks terdifinisi }  { F.S. Inverse matriks ditulis ke layar pengguna atau ke dalam file } |
| Void solveCofactor ()  { I.S. Matriks terdefinisi }  { F.S Inverse matriks terbentuk dengan metode kofaktor } |
| Void solveGaussJordan ()  { I.S. Matriks terdefinisi }  { F.S. Inverse matriks terbentuk dengan metode gauss jordan } |

**3.3. Class DeterminanSolver**

**3.3.1. Atribut**

|  |
| --- |
| Matriks mat  {Matriks yang akan dicari determinannya} |
| double det  {Nilai determinan yang didapat dari mat} |
| boolean toFile  {Nilai boolean untuk menandakan output program ke file atau layar} |

**3.3.2. Method**

|  |
| --- |
| DeterminanSolver ()  { Menerima input matriks dari file atau keyboard pengguna } |
| Void printDet ()  { I.S. Determinan matriks terdefinisi }  { F.S. Determinan matriks ditulis ke layar pengguna atau ke dalam file } |
| Void solveCofactor ()  { I.S. Matriks terdefinisi }  { F.S. Determinan matriks dihasilkan dengan metode ekspansi kofaktor } |
| Void solveRowRed ()  { I.S. Matriks terdefinisi }  { F.S. Determinan matriks dihasilkan dengan metode reduksi baris } |

**3.4. Class SPLSolver**

**3.4.1. Atribut**

|  |
| --- |
| Matriks mat  {Matriks augmented yang akan dicari solusinya} |
| double[] solutions  {Solusi dari SPL} |
| boolean toFile  {Nilai boolean untuk menandakan output program ke file atau layar} |

**3.4.2. Method**

|  |
| --- |
| SPLSolver ()  { Menerima input SPL dari file atau keyboard pengguna } |
| Void printSols ()  { I.S. Solusi SPL terdefinisi dan merupakan solusi unik }  { F.S. Solusi SPL dituliskan ke layar pengguna atau ke dalam file } |
| Void printParametrik ()  { I.S. Solusi SPL terdefinisi dan merupakan solusi banyak }  { F.S. Solusi parametrik SPL dituliskan ke layar pengguna atau ke dalam file } |
| Void cramer ()  { I.S. Matriks SPL terdefinisi }  { F.S. Terbentuk solusi SPL menggunakan metode cramer } |
| Void gauss ()  { I.S. Matriks SPL terdefinisi }  { F.S. Terbentuk solusi SPL menggunakan metode gauss } |
| Void gaussJordan ()  { I.S. Matriks SPL terdefinisi }  { F.S. Terbentuk solusi SPL menggunakan metode gauss jordan } |
| Void inverse ()  { I.S. Matriks SPL terdefinisi }  { F.S. Terbentuk solusi SPL menggunakan metode inverse } |

**3.5. Class LinearRegression**

**3.5.1. Atribut**

|  |
| --- |
| Matriks mat  {Matriks yang berisi SPL *Normal Estimation Equation for Multiple Linear Regression*} |
| double[] xk  {Nilai-nilai xk untuk menghitung taksiran} |
| boolean toFile  {Nilai boolean untuk menandakan output program ke file atau layar} |

**3.5.2. Method**

|  |
| --- |
| LinearRegression ()  { Menerima input titik-titik dari file atau keyboard pengguna dan menyimpannya ke dalam matriks } |
| Void LRSolve ()  { I.S. Matriks regresi linear terdefinisi }  { F.S. Terbentuk solusi regresi linier dengan metode gauss pada matriks dan menuliskannya ke layar pengguna atau ke dalam file } |

**3.6. Class Interpolasi**

**3.6.1. Atribut**

|  |
| --- |
| Matriks mat  {Matriks yang berisi SPL dari persamaan polinom} |
| double x  {Nilai x yang akan ditaksir hasilnya} |
| boolean toFile  {Nilai boolean untuk menandakan output program ke file atau layar} |

**3.6.2. Method**

|  |
| --- |
| Interpolasi ()  { Menerima input titik-titik dari file atau keyboard pengguna dan menyimpannya ke dalam matriks } |
| Void solve ()  { I.S. Matriks berisi titik-titik input dari file atau keyboard pengguna terdefinisi }  { F.S. Terbentuk solusi interpolasi dengan metode gauss pada matriks dan menuliskannya ke layar pengguna atau ke dalam file } |

**3.7 Garis Besar Program**

Secara umum, program akan menampilkan menu pilihan, pengguna kemudian memasukkan pilihannya dalam bentuk angka, setelah itu pengguna akan diminta untuk memberikan data-data yang diperlukan, dapat berupa file ataupun masukan dari keyboard. Setelah itu, program akan melakukan perhitungan sesuai dengan masukan pengguna, misal menghitung determinan sebuah matriks. Setelah itu program akan mengeluarkan hasil sesuai dengan pilihan, keluaran dapat ditampilkan di layar ataupun juga ke dalam sebuah file.

**BAB IV**

**EKSPERIMEN**

|  |  |  |
| --- | --- | --- |
| **1** | **a** |  |
| **b** |  |
| **c** |  |
| **d** | n = 6 :    n = 10 : |
| **2** | **a** |  |
| **b** |  |
| **3** | **a** |  |
| **b** |  |
| **4** | |  |
| **5** | |  |
| **6** | | 25/05/2020 :  30/08/2020 :  15/09/2020:    17/08/2020: |
| **7** | |  |
| **8** | |  |

**BAB V**

**KESIMPULAN, SARAN, DAN REFLEKSI**

**5.1. Kesimpulan**

Setelah melalui proses pengerjaan dan pengujian, program untuk menghitung sistem persamaan linier, determinan, dan aplikasinya berhasil dibuat menggunakan bahasa pemrograman Java. Secara umum, program dapat menerima input dari pengguna melalui *keyboard* dan melalui file. Seluruh fitur berhasil diimplementasikan pada program yang terdiri atas:

1. Memecahkan sistem persamaan linier menggunakan metode eliminasi Gauss, eliminasi Gauss-Jordan, matriks balikan, dan kaidah Cramer.
2. Melakukan pembalikan matriks dan menghitung determinan menggunakan metode reduksi baris dan ekspansi kofaktor.
3. Menghitung hasil fungsi menggunakan interpolasi polinom.
4. Melakukan regresi linear berganda dengan metode *normal equation* dan menghitung persamaan yang didapat menggunakan metode Gauss.

Sebagai tambahan, penggunaan GUI untuk interaksi dengan pengguna juga berhasil diimplementasikan pada proses input menggunakan file.

**5.2 Saran**

Program dapat diuji lebih lanjut dengan memberikan lebih banyak kasus uji yang beragam. Selain itu, modularitas program juga dapat ditingkatkan dengan membuat fungsi atau prosedur untuk lebih banyak subrutin. Penggunaan GUI juga dapat diimplementasikan untuk seluruh program demi meningkatkan *user-friendliness*.

**5.3 Refleksi**

Kondisi pembelajaran daring saat ini membuat penulis harus bekerja secara *remote* yang membutuhkan beberapa adaptasi. Meskipun tidak dapat bertemu langsung, penulis tetap dapat berkomunikasi satu sama lain dengan baik dan mengerjakan tugas relatif lancar. Penggunaan GitHub untuk berkolaborasi juga sangat membantu dalam pengerjaan tugas, dan mungkin dapat diimplementasikan pula untuk tugas-tugas berikutnya. Sebagai penutup, penulis berharap situasi pandemi ini dapat berlalu dalam waktu singkat dan kita semua dapat bertemu dan berinteraksi kembali secara langsung.

**DAFTAR PUSTAKA**

Munir, Rinaldi. 2020. IF2123 Aljabar Geometri [http://informatika.stei.itb.ac.id/~rinaldi.munir /AljabarGeometri/2020-2021/algeo20-21.htm](http://informatika.stei.itb.ac.id/~rinaldi.munir%20/AljabarGeometri/2020-2021/algeo20-21.htm). Diakses 23 September 2020

<https://stackoverflow.com/>, diakses 22 September 2020 hingga 1 Oktober 2020

<https://www.geeksforgeeks.org/>, diakses 24 September 2020 hingga 30 September 2020

<https://www.w3schools.com/java/>, diakses 19 September 2020 hingga 1 Oktober 2020