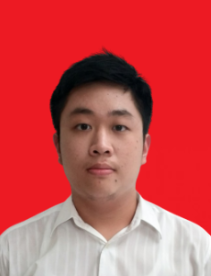
**LAPORAN TUGAS BESAR I**

**IF2123**

**ALJABAR LINEAR dan GEOMETRI**



Disusun oleh :

13519163 - Alvin Wilta

13519199 - Christian Gunawan

13519215 - Leonard Matheus

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

BANDUNG

2020

# **DAFTAR ISI**

# Daftar Isi …………………………………………………………………………………………………. 1

# Bab I Deskrpsi Masalah ……………………………………………………………………………….. 2

# 1.1 Abstraksi................................................................................................................. 2

# 1.2 Interpolasi Polinom................................................................................................. 3

# 1.3 Regresi Linear Berganda................................................................................................................. 4

# Bab II Teori Singkat

# 2.1 Operasi Baris Elementer, Gauss, dan Gauss-Jordan ………………………..……… 5

# 2.2 Determinan …………………….……………………………………………………..…... 5

# 2.3 Matriks Balikan ………...………………………………………………………………… 6

# 2.4 Matriks Adjoint ……………………………………………………………………..…….. 6

# 2.5 Matriks Kofaktor ……………………………………………………………………..…… 6

# 2.6 Metode Cramer ………………………………………………………………………...... 7

# 2.7 Interpolasi Polinom ………………………………………………………………...…….. 7

# 2.8 Regresi Linear Berganda …………………………………………………………..…... 7

# Bab III Implementasi …………………………………………………………………………………... 8

# Bab IV Eksperimen ……………………………………………………………………………………. 11

# Bab V Kesimpulan, Saran, Refleksi ………………………………………………………………… 23

# 5.1 Kesimpulan ......................................................................................................... 23

# 5.2 Saran................................................................................................................... 23

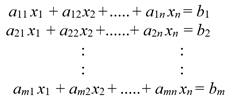
# 5.3 Refleksi................................................................................................................ 23

# Referensi ……………………...…………………………………………………………………..….. 24

# **BAB I. DESKRIPSI MASALAH**

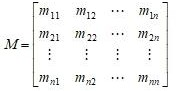
**1.1.** **Abstraksi**

Sistem persamaan linier (SPL) *Ax* = *b* dengan *n* peubah (*variable*) dan *m* persamaan adalah berbentuk

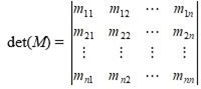
**

yang dalam hal ini *xi* adalah peubah, *aij* dan *bi* adalah koefisien ∈ R. 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, atau hanya satu (unik/tunggal).

Sebuah matriks *M* berukuran *n* × *n*

**

determinannya adalah



Determinan matriks *M* berukuran *n* × *n* dapat dihitung dengan beberapa cara: reduksi baris dan ekspansi kofaktor.

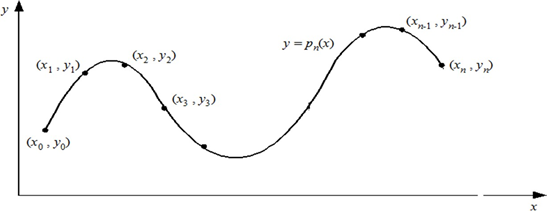
SPL memiliki banyak aplikasi dalam bidang sains dan rekayasa, dua diantaranya diterapkan pada tugas besar ini, yaitu interpolasi polinom dan regresi linier.

# 

# 

# **1.2.** **Interpolasi Polinom**

Persoalan interpolasi polinom adalah sebagai berikut: Diberikan *n*+1 buah titik berbeda, (*x*0, *y*0), (*x*1, *y*1), ..., (*xn*, *yn*). Tentukan polinom *pn*(*x*) yang menginterpolasi (melewati) semua titik-titik tersebut sedemikian rupa sehingga *yi* = *pn*(*xi*) untuk *i* = 0, 1, 2, …, *n*.

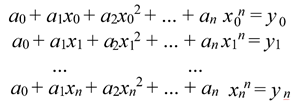


Setelah polinom interpolasi *pn*(*x*) ditemukan, *pn*(*x*) dapat digunakan untuk menghitung perkiraan nilai *y* di sembarang titik di dalam selang [*x*0, *xn*].

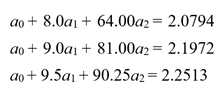
Polinom interpolasi derajat *n* yang menginterpolasi titik-titik (*x*0, *y*0), (*x*1, *y*1), ..., (*xn*, *yn*). adalah berbentuk *pn*(*x*) = *a*0 + *a*1*x* + *a*2*x*2 + … + *anxn*. Jika hanya ada dua titik, (*x*0, *y*0) dan (*x*1, *y*1), maka polinom yang menginterpolasi kedua titik tersebut adalah *p*1(*x*) = *a*0

+ *a*1*x* yaitu berupa persamaan garis lurus. Jika tersedia tiga titik, (*x*0, *y*0), (*x*1, *y*1), dan (*x*2, *y*2), maka polinom yang menginterpolasi ketiga titik tersebut adalah *p*2(*x*) = *a*0 + *a*1*x*

+ *a*2*x*2 atau persaman kuadrat dan kurvanya berupa parabola. Jika tersedia empat titik, (*x*0, *y*0), (*x*1, *y*1), (*x*2, *y*2), dan (*x*3, *y*3), polinom yang menginterpolasi keempat titik tersebut adalah *p*3(*x*) = *a*0 + *a*1*x* + *a*2*x*2 + *a*3*x*3, demikian seterusnya. Dengan cara yang sama kita dapat membuat polinom interpolasi berderajat *n* untuk *n* yang lebih tinggi asalkan tersedia (*n*+1) buah titik data. Dengan menyulihkan (*xi*, *yi*) ke dalam persamaan polinom *pn*(*x*) = *a*0 + *a*1*x* + *a*2*x*2 + … + *anxn* untuk *i* = 0, 1, 2, …, *n*, akan diperoleh *n* buah sistem persamaan lanjar dalam *a*0, *a*1, *a2*, …, *an*

**

Solusi sistem persamaan lanjar ini, yaitu nilai *a*0, *a*1, …, *an*, diperoleh dengan menggunakan metode eliminasi Gauss yang sudah anda pelajari. Sebagai contoh, misalkan diberikan tiga buah titik yaitu (8.0, 2.0794), (9.0, 2.1972), dan (9.5, 2.2513). Tentukan polinom interpolasi kuadratik lalu estimasi nilai fungsi pada *x* = 9.2. Polinom kuadratik berbentuk *p*2(*x*) = *a*0 + *a*1*x* + *a*2*x*2. Dengan menyulihkan ketiga buah titik data ke dalam polinom tersebut, diperoleh sisteM persamaan lanjar yang terbentuk adalah

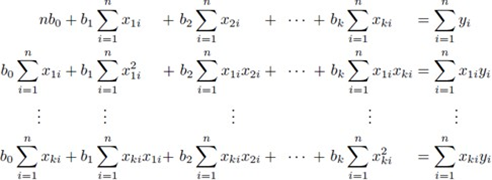


Penyelesaian sistem persamaan dengan metode eliminasi Gauss menghasilkan *a*0 = 0.6762, *a*1 = 0.2266, dan *a*2 = -0.0064. Polinom interpolasi yang melalui ketiga buah titik tersebut adalah *p*2(*x*) = 0.6762 + 0.2266*x* - 0.0064*x*2. Dengan menggunakan polinom ini, maka nilai fungsi pada *x* = 9.2 dapat ditaksir sebagai berikut: *p*2(9.2) = 0.6762 + 0.2266(9.2) - 0.0064(9.2)2 = 2.2192.

# **1.3.** **Regresi Linier Berganda**

Regresi Linear (akan dipelajari lebih lanjut di Probabilitas dan Statistika) merupakan salah satu metode untuk memprediksi nilai selain menggunakan Interpolasi Polinom. Meskipun sudah ada rumus jadi untuk menghitung regresi linear sederhana, terdapat rumus umum dari regresi linear yang bisa digunakan untuk regresi linear berganda, yaitu.

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



Sistem persamaan linier tersebut diselesaikan dengan menggunakan metode eliminasi Gauss.

# 

# 

# **BAB II. TEORI SINGKAT**

**2.1. Operasi Baris Elementer, Gauss, dan Gauss Jordan**

Ketika dihadapi masalah yang berkaitan dengan sistem persamaan linear terutama yang menggunakan banyak peubah, maka hal pertama yang dapat digunakan untuk menyederhanakan permasalahan adalah dengan mengubah sistem persamaan linear yang ada ke dalam bentuk matriks. Suatu persamaan linear biasanya juga tidak didapatkan secara langsung tetapi melalui penyederhanaan dari permasalahan yang terjadi dalam kehidupan sehari – hari. Setelah diubah ke bentuk matriks, maka matriks tersebut diubah ke bentuk matriks dalam bentuk eselon baris tereduksi untuk mendapatkan penyelesaian dari SPL.

Prosedur untuk mendapatkan matriks eselon baris tereduksi biasa disebut sebagai eliminasi Gauss– Jordan . Pada proses eliminasi tersebut operasi – operasi yang digunakan disebut operasi baris elementer. Dalam operasi baris elementer ini ada beberapa operasi yang dapat digunakan , yaitu :

a. Mengalikan suatu baris dengan konstanta tak nol

b. Mempertukarkan dua buah baris

c. Menambahkan kelipatan suatu baris ke baris lainnya.

Dengan menggunakan operasi baris elementer , maka matriks eselon baris tereduksi yang didapatkan akan ekuivalen dengan matriks awalnya sehingga penyelesaian untuk matriks eselon baris tereduksi juga merupakan penyelesaian untuk matriks awalnya. Matriks awal yang dimaksud adalah matriks diperbesar.

**2.2.Determinan**

Determinan adalah suatu bilangan riil yang diperoleh dari suatu proses dengan aturan tertentu terhadap matriks bujur sangkar. Fungsi determinan di A, disebut atau ditulis det A adalah jumlah semua perkalian elementer dari A.Notasi / simbol lainnya yang banyak dipakai untuk menyatakan determinan dari A, selain det A adalah ½A½.

Berikut ini adalah sifat determinan:

· Nilai suatu determinan tidak berubah jika baris-barisnya ditulis sebagai kolom-kolomnya, dalam urutan yang sama.

· Jika semua unsur dari satu baris atau kolom dari suatu determinan dikalikan oleh faktor k yang sama, maka nilai dari determinan yang baru, sama dengan k kali nilai determinan yang diketahui.

· Jika unsur dalam suatu baris ( atau suatu kolom ) dari suatu determinan adalah nol, maka nilai determinan itu sama dengan nol

· Jika setiap unsur dalam suatu baris atau kolom dari suatu determinan dinyatakan sebagai suatu binomial, maka determinan itu dapat ditulis sebagai jumlah dari dua determinan.

· Jika sembarang dua baris atau kolom determinan dipertukarkan, maka nilai determinan itu dikalikan dengan –1.

· Jika unsur-unsur yang berkaitan dari dua baris atau kolom suatu determinan adalah sebanding, maka nilai determinan itu sama dengan nol.

· Nilai suatu determinan tidak berubah jika unsur-unsur dari suatu baris atau kolom diubah dengan menambahkan pada unsur-unsur tadi sembarang konstanta kali unsur-unsur yang berpadanan dari sembarang baris ( atau kolom secara berturut-turut) lainnya.

· Untuk sembarang matriks A dan B yang berukuran n x nDet (AB) = det (BA) = det A det B.

**2.3. Matriks Balikan**

Misalkan A matriks bujur sangkar, matriks B yang memenuhi AB = BA = I , disebut sebagai invers dari A. Matriks A yang mempunyai invers disebut sebagai matriks tak singular atau invertible, sedangkan yang tidak mempunyai invers disebut matriks singular.

**2.4. Matriks Adjoint**

Definisi: Jika A sebarang matriks n x n dan Cij adalah kofaktor aij, maka matriks dinamakan matriks kofaktor A Transpose dari matriks kofaktor adalah adjoint (sering ditulis adj(nama\_matriks) Transpose matriks kofaktor A adalah Adjoin A (adj(A)).

**2.5. Matriks Kofaktor**

Misalkan A = (aij ) adalah matriks bujur sangkar maka minor pada entri aij dinyatakan oleh |Mij | dan didefinisikan menjadi determinan sub-matriks, setelah baris ke−i dan kolom ke−j dihapuskan dari A. Bilangan (−1)(1+j) |Mij | dinyatakan oleh Kij dinamakan kofaktor entri aij [5].

**2.6. Metode Cramer**

Jika Ax = b adalah sebuah sistem linear n yang tidak diketahui dan det(A)≠ 0 maka persamaan tersebut mempunyai penyelesaian yang unik dimana Aj adalah matrik yang didapat dengan mengganti kolom j dengan matrik b

**2.7. Interpolasi Polinom**

Interpolasi adalah taksiran harga-harga di antara titik-titik diskrit didalam bentangan data benar-benar tepat dan pendekatannya adalah mencari kurva tunggal atau sederetan kurva yang tepat melalui titik titik tersebut (Kristoko Dwi Hartono 2006).

Interpolasi polinomial Lagrange hampir sama dengan polinomial Newton, tetapi tidak menggunakan bentuk pembagian beda hingga. Interpolasi polinomial Lagrange dapat diturunkan dari persamaan Newton. Interpolasi Lagrange diterapkan untuk mendapatkan fungsi polinomial P (x) berderajat tertentu yang melewati sejumlah titik data. Misalnya, kita ingin mendapatkan fungsi polinomial berderajat satu yang melewati dua buah titik yaitu (x0, y0) dan (x1, y1).

**2.8.Regresi Linier Berganda**

Analisis regresi merupakan metode analisis yang dapat digunakan untuk menganalisis data dan mengambil kesimpulan yang bermakna tentang hubungan ketergantungan variabel terhadap variabel lainnya. Hubungan yang didapat pada umumnya dinyatakan dalam bentuk persamaan matematika yang menyatakan hubungan antara variabel bebas (independent variable) dan variabel tak bebas (dependent variable) dalam bentuk persamaan sederhana (Drapper dan Smith, 1992).

Pengujian kenormalan digunakan untuk mengetahui apakah residual yang didapatkan dalam regresi linier berganda metode kuadrat terkecil mengikuti pola distribusi normal atau tidak. Uji yang dapat digunakan adalah uji Kolmogorov Smirnov.

**BAB III. IMPLEMENTASI PROGRAM**

Source code dari program yang dibuat terdiri atas dua file .java, yakni MainProg.java dan matriks.java. MainProg.java memuat method main(), sehingga program akan dijalankan melalui file tersebut. File matriks.java memuat deklarasi class matriks beserta konstruktor dan primitif-primitif (fungsi dan metode) yang dapat dilakukan kepada objek yang berasal dari class matriks matriks.java dan MainProg.java. Pada matriks.java didefinisikan beberapa atribut matriks seperti baris, kolom, dan elemen-elemen pada matriks tersebut, kemudian didefinisikan beberapa metode dan fungsi sebagai berikut :

Void matriks() = Konstruksi Matriks

Public int LeftestOneKoef(int a) = Fungsi yang mengembalikan indeks angka

Public double Elmt(int row, int col) = Fungsi yang mengembalikan matriks yang dituju

Void TambahBaris() = Menambahkan baris

Void TukarBaris() = Menukar baris

Void KaliBaris() = Mengalikan baris

Float pangkat() = Memangkatkan baris

FUNGSI POKOK

Void bacaFileExtSPL() = membaca SPL dari file external

Void bacaFileExtInterpolasi() = membaca matriks interpolasi dari file external

Void bacaFileExtDeterminan() = membaca determinan dari file external

Void bacaFileBalikan() = membaca balikan matriks dari prosedur

bacaUkuranMatriks

Void bacaUkuranMatriks() = menerima input banyaknya baris dan

banyaknya kolom dari suatu matriks

Void bacaUkuranMatriksInterpolasi() = membaca baris dari matriks interpolasi

Void bacaMatriksInterpolasi() = membaca isi matriks interpolasi

Void bacaMatriksSPLExt() = membaca matriks SPL dari file external

Void bacaFileExtBalikan() = membaca matriks balikan dari file external

Void bacaMatriksSPLGauss() = membaca matriks SPL dengan perhitungan

Gauss

Void bacaMatriksSPLGaussJordan() = membaca matriks SPL dengan perhitungan

Gauss-Jordan

Void bacaMatriksBalikanSPL() = membaca matriks SPL dengan matriks

balikan

Void bacaMatriksBalikan() = membaca matriks dengan matriks balikan

Void bacaMatriksRegresi() = membaca matriks dengan regresi

Boolean is\_square() = mengembalikan true jika matriks berukuran

sama

Boolean isBarNol(int i) = mengembalikan true jika baris i matriks

bukan bernilai nol

Boolean isKolNol(int i,int j) = mengembalikan true jika kolom j matriks dari

baris i sampai baris : 0

Float pangkat(float x,int i) = mengembalikan nilai x pangkat i

Float kaliDiagonal() = mengembalikan hasil perkalian diagonal

Matriks

Void tulisMatriks() = menuliskan matriks pada layar

Void tukarBaris(int i,int j) = menukar baris i dan kolom j pada matriks

Int indeksTakNol(int i,int j) = mengeluarkan indeks pertama tak bernilai nol

dari matriks pada kolom j dan dimulai baris i

Void buatLeadingOne(int i,int j) = menghasilkan matriks yang memiliki leading

Point 1 pada baris i

Void buatKolomNolBawah(int j, int i) = membuat kolom j nol dimulai dari baris ke i+1

Void buatKolomNolAtas(int i,int j) = membuat kolom j berisi nol diatas indeks i

Int indeksPivot(int i) = mengembalikan indeks pivot baris i

Void Transpose() = menghasilkan matriks transpose

Void BacaDeterminant() = membaca nilai determinan

Void matriksInterpolasi() = mengubah matriks interpolasi dari user

menjadi matriks interpolasi untuk dicari solusi

Void matriksInterpolasiExt() = mengubah matriks interpolasi dari file

eksternal menjadi matriks interpolasi

Void matriksRegresi() = mengubah matriks Multiple Linear Regression menjadi array berisikan parameter dari persamaan umum regresi linear

void matriksRegresiExt() = membaca matriks Multiple Linear regression

menjadi array dari file eksternal

void tulisInterpolasi() = menuliskan nilai dari metode interpolasi

void tulisInterpolasiGauss() = menuliskan nilai dengan metode gauss

void Gauss() = mengubah matriks menjadi Row Echelon

Form

void tulisGauss() = menghasilkan penyelesaian matriks dengan

metode Gauss

void GaussJordan() = mengubah matriks menjadi Row Echelon

Form tereduksi

void tulisGaussJordan() = menghasilkan penyelesaian matriks dengan

metode Gauss-Jordan

Void DeterminanReduksi() = menghasilkan determinan dengan metode

reduksi baris

Void DeterminanKofaktor() = menghasilkan determinan dengan metode

kofaktor

Void InverseReduksi() = menghasilkan invers matriks dengan metode

reduksi baris

Void InverseKofaktor() = menghasilkan invers matriks dengan metode

kofaktor

Void tulisInverseReduksi() = menuliskan matriks hasil invers dengan metode reduksi baris

Void tulisInverseKofaktor() = menuliskan matriks hasil invers dengan metode kofaktor

float InverseMatriksSPL() = mengembalikan nilai invers matriks SPL

void tulisInverseMatriksSPL() = menuliskan hasil invers matriks SPL

float Crammer() = melakukan operasi cramer

void tulisCramer() = menghasilkan penyelesaian matriks kaidah

cramer

Void buatAdjoin() = prosedur untuk menghasilkan matriks adjoin

static void MenuUtama() :

Menampilkan Menu Utama berupa:

· Sistem Persamaaan Linier

· Determinan

· Matriks balikan

· Interpolasi Polinom

· Regresi linier berganda

· Keluar

static void MenuPilihanSPL()

Menampilkan Menu Pilihan SPL Berupa:

static void MenuPilihanInterpolasi

static void MenuPilihanRegresi()

static void MenuPilihanDeterminan

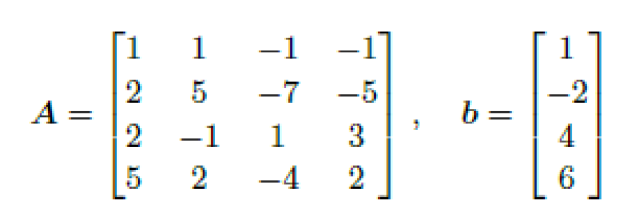
static void MenuPilihanInverse()

static void MenuInput()

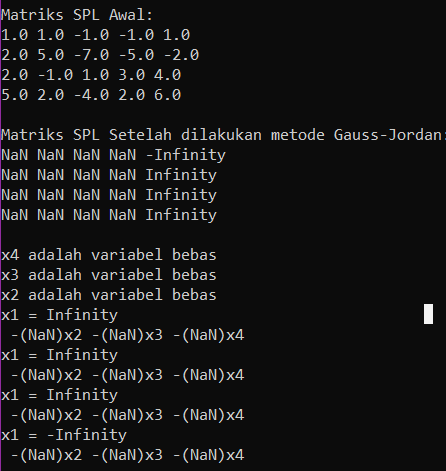
static void Keluar()

**BAB IV. Eksperimen**

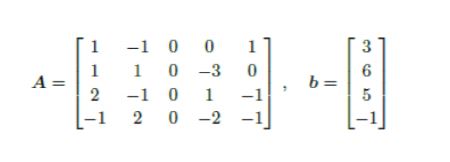
1. Sistem Persamaan Linear(SPL) Ax=b
2. Dengan soal sebagai berikut :

****

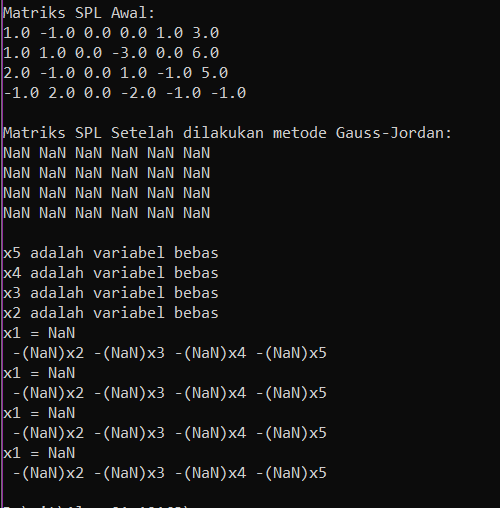
Hasil evaluasi program :



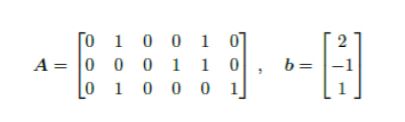
1. Dengan soal sebagai berikut ::



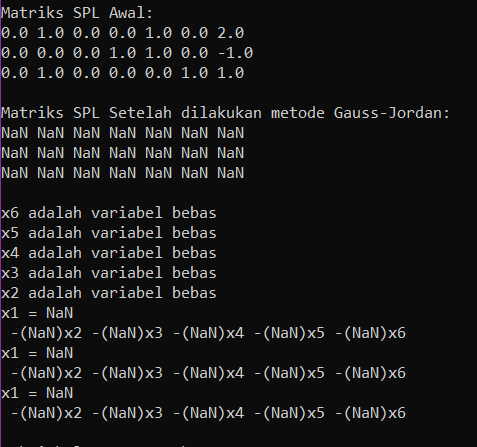
Hasil evaluasi program :



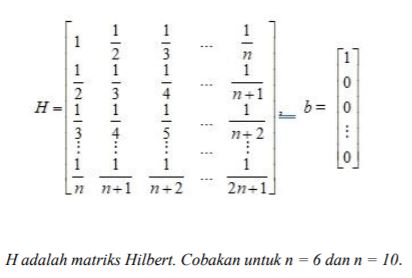
1. Dengan soal sebagai berikut ::



Hasil evaluasi program :

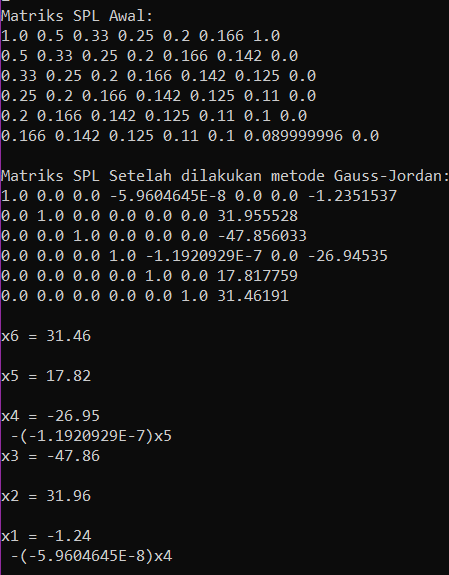


1. Dengan soal sebagai berikut ::

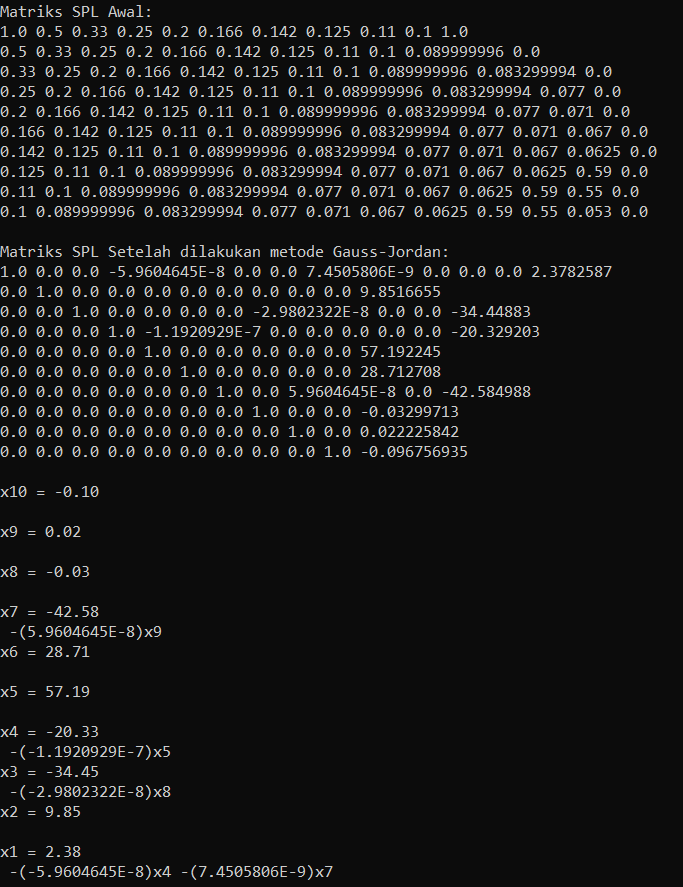


Hasil evaluasi program :

N = 6

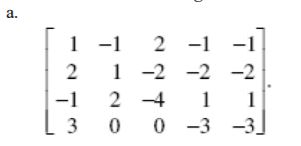


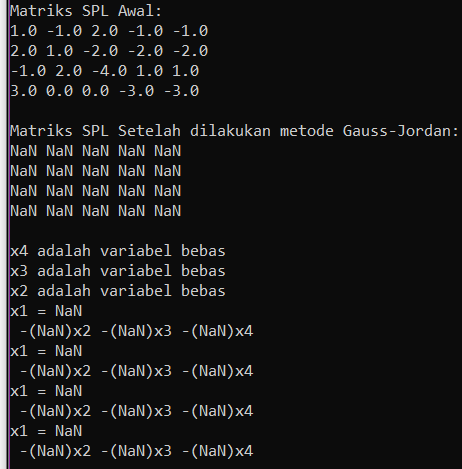
n= 10



2. SPL berbentuk matriks *Augmented*

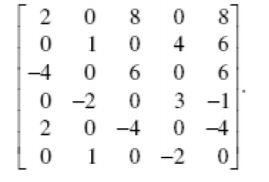
1. Dengan soal sebagai berikut :



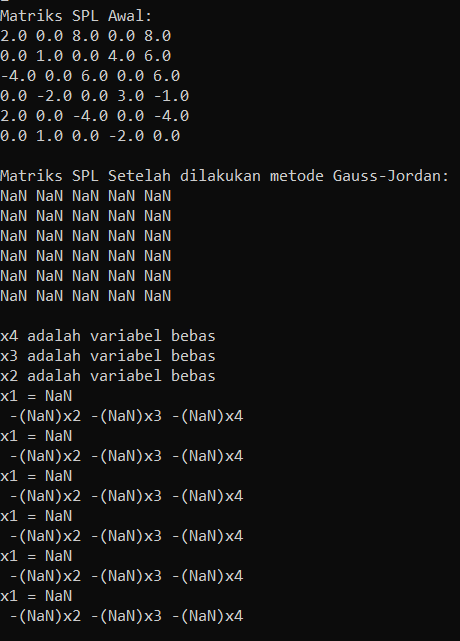


Hasil evaluasi program :

1. Dengan soal sebagai berikut :



Hasil evaluasi program :



3. SPL dalam bentuk persamaan-persamaan :

1. Dengan soal sebagai berikut :

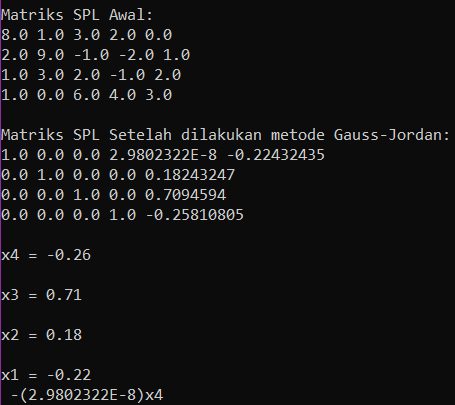
8x1 + x2 + 3x3 + 2x4 = 0

2x1 + 9x2 - x3 - 2x4 = 1

x1 + 3x2 + 2x3 - x4 = 2

x1 + 6x3 + 4x4 = 3

Hasil evaluasi program :



4. Studi Kasus : interpolasi

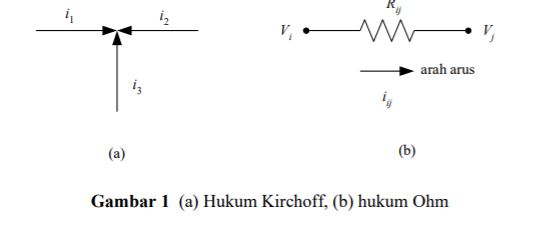
Dalam sebuah rangkaian listrik berlaku hukum arus Kirchoff yang menyatakan bahwa jumlah aljabar dari semua arus yang memasuki suatu simpul adalah nol, yakni :

Σ i = 0

Selain itu, berlaku pula hukum Ohm yang menyatakan bahwa besar arus yang melalui suatu resistor adalah sebesar :

iij = (𝑽𝒊−𝑽𝒋)/𝑹𝒊

yang dalam hal ini V adalah tegangan dan R adalah tahanan.



Dari hukum Kirchoff diperoleh persamaan-persamaan sebagai berikut

i12 + i52 + i32 = 0

i65 – i52 – i54 = 0

i43 – i32 = 0

i54 – i43 = 0

Dari hukum OHm diperoleh persamaan-persamaan sebagai berikut

i32 R32 - V3 + V2 = 0

i43 R43 - V4 + V3 = 0

i65 R65 + V5 = V6

i12 R12 + V2 = V1

i54 R54 - V5 + V4 = 0

i52 R52 - V5 + V2 = 0

Diketahui :

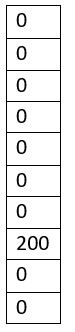
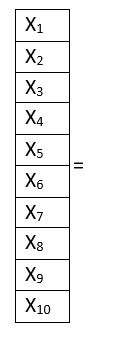
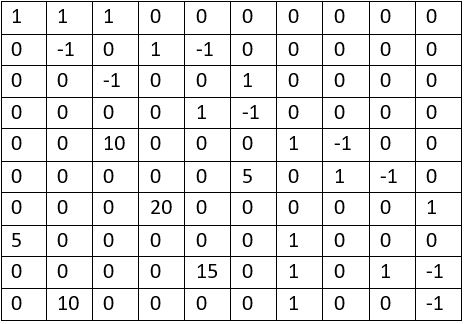
R12 = 5 ohm , R52 = 10 ohm , R32 = 10 ohm R65 = 20 ohm , R54 = 15 ohm , R14 = 5 ohm. V1 = 200 volt , V6 = 0 volt.

Maka nilai dari variabel i12, i52, i32, i65, i54, i13, V2, V3, V4, V5 dapat dicari.

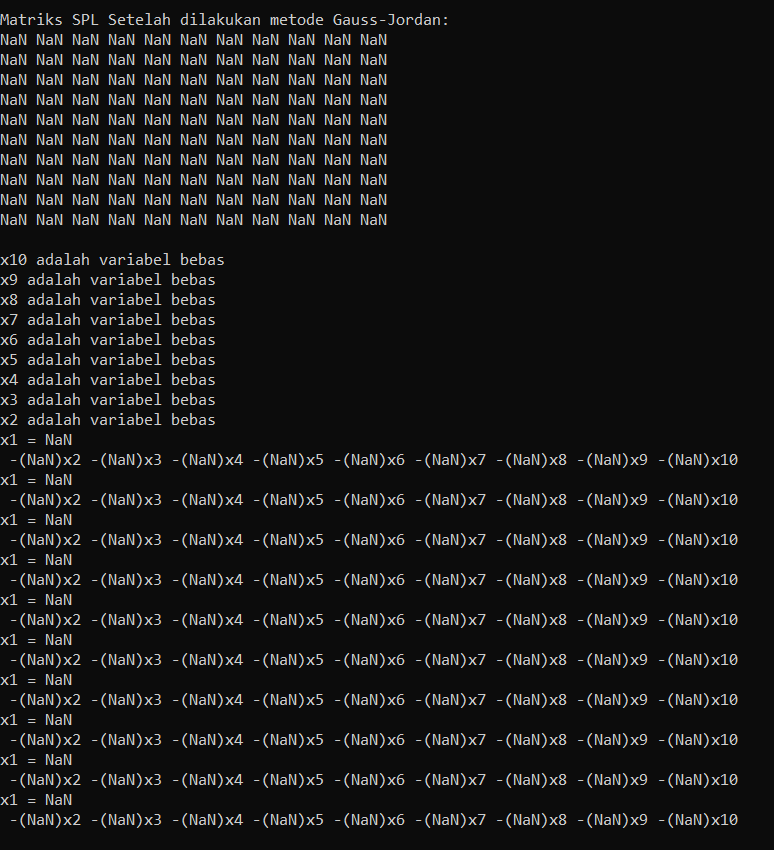
Dengan mengasumsikan variabel-variabel diatas menjadi sebagai berikut :

i12 = x1, i52 = x2, i32 = x3, i65 = x4, i54 = x5, i43 = x6, V2 = x7, V3 = x8, V4 = x9, V= = x10

Maka kita dapat memasukkan asumsi diatas kedalam bentuk persamaan matriks 10x10 seperti berikut



Hasil keluaran program dengan SPL diatas adalah :



5. Interpolasi

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).



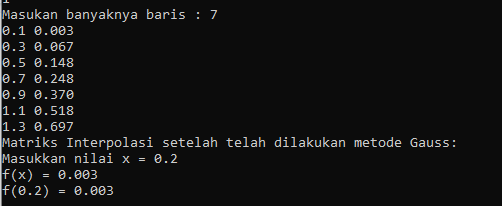
Lakukan pengujian pada nilai-nilai default berikut:

x = 0.2 f(x) = 0.003

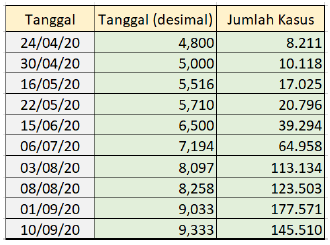
x = 0.55 f(x) = ?

x = 0.85 f(x) = ?

x = 1.28 f(x) = ?



6. Jumlah kasus positif Covid-19 di Indonesia semakin bertambah dari hari ke hari. Di bawah ini diperlihatkan jumlah kasus Covid-19 di Indonesia mulai dari tanggal 24 April 2020 hingga 10 September 2020:



Tanggal (desimal) adalah tanggal yang sudah diolah ke dalam bentuk desimal 3 angka di belakang koma dengan memanfaatkan perhitungan sebagai berikut:

tanggal(desimal) = bulan + (tanggal / jumlah hari pada bulan tersebut)

berdasarkan data diatas, prediksi jumlah kasus pada tanggal di bawah ini adalah:

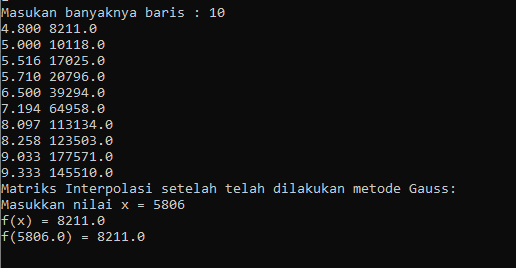
a. 25/05/20 = Tanggal (5.806), f(tanggal) = 8211.0

b. 30/08/20 = Tanggal (9.000), f(tanggal) = 176873

c. 15/09/20 = Tanggal (9.500), f(tanggal) = 68216

d. Input user ( asumsi 30/4/20) = Tanggal (5.000), f(tanggal) = 10118

Hasil dari interpolasi adalah sebagai berikut:

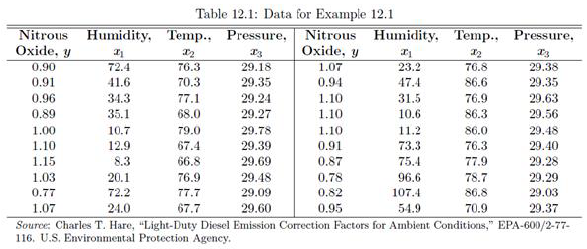


7. Penyederhanaan fungsi f(x) dengan polinom interpolasi derajat n di dalam selang [0, 2]. Sebagai contoh, jika n = 5, maka titik-titik x yang diambil di dalam selang [0, 2] berjarak h = (2 – 0)/5 = 0.4.



maka f(x) yang baru bernilai:

8. Diberikan sekumpulan data sesuai pada tabel berikut ini.



Gunakan Normal Estimation Equation for Multiple Linear Regression 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.

Dari data-data tersebut, apabila diterapkan Normal Estimation Equation for Multiple Linear Regression, maka diperoleh sistem persamaan linear sebagai berikut.

20b0 + 863.1b1 + 1530.4b2 + 587.84b3 = 19.42

863.1b0 + 54876.89b1 + 67000.09b2 + 25283.395b3 = 779.477

1530.4b0 + 67000.09b1 + 117912.32b2 + 44976.867b3 = 1483.437

587.84b0 + 25283.395b1 + 44976.867b2 + 17278.5086b3 = 571.1219

Prediksi dengan menggunakan Langrange Multiple Linear Regression:

b0 = 43.270015

b1 = 1810.861310

b2 = 3315.578331

b3 = 1278.063176

**BAB V KESIMPULAN, SARAN, REFLEKSI**

**5.1 Kesimpulan**

Kami berhasil membuat suatu program operasi matriks dengan baik dan sangat memahami esensi dari adanya tugas besar ini. Selain itu, kami juga dapat menghargai usaha dan kerja keras kami bertiga dalam menyelesaikan tugas besar ini.

**5.2 Saran**

Program ini bisa ditambah dengan fitur GUI sehingga memudahkan pengguna dalam memasukkan angka matriks ke dalam program tanpa harus mengetik satu per-satu.

**5.3 Refleksi**

Dengan adanya tugas besar ini, kami perlu mempelajari bahasa pemrograman yang belum familiar dengan kami, yakni Java. Dengan waktu yang terbilang singkat, kami merasa pekerjaan masih dapat diperbaiki dengan lebih banyak belajar. Walaupun kurangnya pemahaman kami, kami tetap memberikan usaha terbaik kami untuk menyelesaikan tugas besar yang diberikan kepada kami.

**REFERENSI**

Howard, A. (2013). Elementary Linear Algebra 11th Edition.

<http://wwwf.imperial.ac.uk/~das01/GSACourse/Regression.pdf>

<http://home.iitk.ac.in/~shalab/regression/Chapter3-Regression-MultipleLinearRegressionModel.pdf>

<http://etheses.uin-malang.ac.id/6753/1/98120662.pdf>