**Laporan TugasBesar IA**

**Aljabar Geometri**

**“Aplikasi Aljabar Lanjar pada Metode Numerik”**

**- Davin Prasetya (13514003) –**

**- Nathan James Runtuwene (13514083) –**

**- Christian Anthony S. (13514085) –**

****

**Institut Teknologi Bandung**

**29 September 2015**

# Daftar Isi

[Daftar Isi 2](#_Toc431507319)

[Bab 1 Deskripsi Masalah 3](#_Toc431507320)

[Bab 2 Teori Singkat 4](#_Toc431507321)

[2.1 Eliminasi Gauss 4](#_Toc431507322)

[2.2 Eliminasi Gauss-Jordan 4](#_Toc431507323)

[2.3 Tatancang Pemorosan 4](#_Toc431507324)

[2.4 Interpolasi 4](#_Toc431507325)

[Bab 3 Design Class 5](#_Toc431507326)

[1.class InputReader 5](#_Toc431507327)

[2.class Matrix 5](#_Toc431507328)

[3. class Interpolasi 5](#_Toc431507329)

[4.class SistemPersamaanLinear 6](#_Toc431507330)

[5. class MainProgram 6](#_Toc431507331)

[Bab 4 Eksperimen 7](#_Toc431507332)

[Soal A1 7](#_Toc431507333)

[Soal A2 7](#_Toc431507334)

[Soal A3 7](#_Toc431507335)

[Soal B1 7](#_Toc431507336)

[Soal B2 8](#_Toc431507337)

[Soal B3 8](#_Toc431507338)

[Soal B4a 8](#_Toc431507339)

[Soal B4b 9](#_Toc431507340)

[Soal B5 9](#_Toc431507341)

# Bab 1 Deskripsi Masalah

Sistem persamaan lanjar (SPL) merupakan permasalahan umum yang ditemui dalam berbagai bidang, terutama fisika dan matematika. Jumlah variabel dan jumlah persamaan yang banyak seringkali menghabiskan banyak waktu untuk menemukan solusinya. Selain itu, ada juga kasus dimana SPL ini tidak menghasilkan suatu solusi atau bahkan banyak solusi. Maka itu, program ini dibuat untuk menyelesaikan sistem persamaan lanjar pada metode numerik sehingga dapat menghemat waktu untuk menemukan solusi dari persamaan yang diterima.

Program dirancang untuk dengan menggunakan bahasa pemrograman Java. Metode penyelesaian yang digunakan adalah metode eliminasi Gauss dan metode eliminasi Gauss-Jordan. Di dalam kedua metode digunakan tatancang pemorosoan untuk mengurangi galat pembulatan.

Selain menyelesaikan SPL yang memiliki solusi unik, banyak solusi dan tidak ada solusi, program juga dibuat untuk memperkirakan sebuah persamaan yang mewakili dari solusi berupa titik-titik penyelesaian (Interpolasi)

# Bab 2 Teori Singkat

## 2.1 Eliminasi Gauss

Metode eliminasi Gauss adalah metode untuk menyelesaikan suatu sistem persamaan lanjar dengan cara merepresentasikan persamaan tersebut ke dalam sebuah matriks. Matriks tersebut kemudian dibuat sedemikian rupa menggunakan operasi operasi dasar matriks sehingga nilai-nilai matriks menjadi lebih sederhana dengan membentuk nol elemen-elemen dibawah maupun atas diagonal utama matriks.

## 2.2 Eliminasi Gauss-Jordan

Metode eliminasi Gauss-Jordan adalah pengembangan dari metode eliminasi gauss dengan cara melanjutkan hasil matriks eliminasi gauss diusahakan sehingga nilai yang tersisa dalam matriks hanyalah nilai yang berada pada diagonal utama matriks saja. Caranya dengan menggunakan metode eliminasi gauss tetapi dengan fokus/acuan yaitu nilai pada kanan bawah matriks.

## 2.3 Tatancang Pemorosan

Tatancang Pemorosan adalah sebuah tatancang untuk menyelesaikan persamaan SPL menggunakan metode eliminasi Gauss atau Gauss-Jordan dengan cara mencari nilai terbesar dalam kolom pertama matriks, kemudian menjadikan nya sebagai pivot dengan menukarkan nilai tersebut dengan baris pertama. Baris tersebutlah yang menjadi acuan pemorosan. Setelah ditukar, matriks kemudian baru diselesaikan dengan eliminasi Gauss atau Gauss-Jordan. Alasan utama digunakan tatancang pemorosan adalah agar solusi SPL lebih akurat karena dapat mengurangi galat pembulatan.

## 2.4 Interpolasi

Interpolasi adalah metode untuk menghasilkan titik titik baru dari titik titik yang tersedia. Titik baru itu bisa didapat dari membuat suatu persamaan terlebih dahulu dari titik-titik tersebut. Untuk membuat persamaannya, dibuatlah suatu matriks sesuai dengan banyak titiknya. Nilai nilai dari matriks tersebut merupakan perpangkatan dari nilai absis dari titik tersebut. Dari matriks kemudian diselesaikan dengan menggunakan metode eliminasi Gauss, Gauss-Jordan ataupun metode lainnya sehingga didapat nilai pengali masing-masing dari perpangkatan titik tersebut. Dari persamaan situ lah maka nilai titik titik baru dapat ditemukan.

# Bab 3 Design Class

|  |  |
| --- | --- |
| 1.class InputReader | |
| Attribut | |
|  | StringTokenizer tokenizer |
|  | BufferedReader in |
| Method | |
|  | void close() |
|  | String nextToken() |
|  | int nextInt() |
|  | long nextLong() |
|  | double nextDouble() |
|  | next nextString() |
|  | char nextChar() |

|  |  |
| --- | --- |
| 2.class Matrix | |
| Attribut | |
|  | double[][] M |
|  | int n,m |
| Method | |
|  | double[][] getM() |
|  | double getEl() |
|  | void eliminateUsingGaussMethod() |
|  | void swapRow |
|  | void eliminateUsingGaussJordanMethod() |
|  |  |
|  |  |

|  |  |
| --- | --- |
| 3. class Interpolasi | |
| Attribut | |
|  | Int jmlTitik |
|  | SistemPersamaanLinear persamaan |
| Method | |
|  | void read(InputReader in) |
|  | void Solve() |
|  | void Write(BufferedWriter writer) |
|  | double findY(double x) |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| 4.class SistemPersamaanLinear | |
| Attribut | |
|  | int jmlVariabel |
|  | int jmlPersamaan |
|  | Matrix matriks |
|  | TreeMap<String,Integer> mapVariabel |
|  | boolean solved |
|  | String[] listVariabel |
| Method | |
|  | matrix getMatriks() |
|  | void setMatriks(Matriks matriks) |
|  | void read(InputReader in) |
|  | void solve() |
|  | void write(BufferedWriter writer) |
|  |  |
|  |  |

|  |  |
| --- | --- |
| 5. class MainProgram | |
| Attribut | |
|  | InputReader mainProgramReader, inputReader |
|  | BufferedWriter outputWriter |
|  | String inputString |
|  | int inputInt |
| Method | |
|  | void launchSettings() |
|  | void launchMenu() |
|  | void launchMatriks() |
|  | void launchSPL() |
|  | void launchInterpolasi() |
|  | void launchAbout() |
|  | void main(Strings[] args) |

# Bab 4 Eksperimen

## Soal A1

0.31 x1 + 0.14 x2 + 0.30 x3 + 0.27 x4 = 1.02

0.26 x1 + 0.32 x2 + 0.18 x3 + 0.24 x4 = 1.00

0.61 x1 + 0.22 x2 + 0.20 x3 + 0.31 x4 = 1.34

0.40 x1 + 0.34 x2 + 0.36 x3 + 0.17 x4 = 1.27

## Soal A2

1 x1 + 3 x2 – 2 x3 + 2 x5 = 0

2 x1 + 6 x2 – 5 x3 + 4 x4 -3 x6 = -1

5 x3 + 10 x4 + 15 x6 = 5

2 z1 + 6 x2 + 8 x4 + 4 x5 + 18 x6 = 6

## Soal A3

0.7071 0 0 -1 -0.8660 0 0 0 0 0

0.7071 0 1 0 0.5 0 0 0 0 -1000

0 1 0 0 0 -1 0 0 0 0

0 0 -1 0 0 0 0 0 0 0

0 0 0 0 0 0 1 0 0.7071 500

0 0 0 1 0 0 0 0 -0.7071 0

0 0 0 0 0.8660 1 0 -1 0 0

0 0 0 0 -0.5 0 -1 0 0 -500

0 0 0 0 0 0 0 0 0.7071 0

## Soal B1

-0.866 F1 + 0.5 F3 = 0

-0.5 F1 - 0.866 F3 = -1000

-0.866 F1 - 1F2 - 1H2 = 0

-0.5 F1 -1 V2 = 0

-1 F2 - 0.5 F3 = 0

-0.866 F3 – 1 V3 = 0

## Soal B2

1 I12 + 1 I52 + 1 I32 = 0

1 I65 – 1 I52 – 1 I54 = 0

1 I43 – 1 I32 = 0

1 I54 – 1 I43 = 0

10 I32 – 1 V3 + 1 V2 = 0

5 I43 – 1 V4 + 1 V3 = 0

20 I65 + 1 V5 = 0

5 I12 + 1 V2 = 0

15 I54 – 1 V5 + 1 V4 = 0

10 I52 – 1 V5 + 1 V2 = 0

## Soal B3

0.1 0.003

0.3 0.067

0.5 0.148

0.7 0.248

0.9 0.370

1.1 0.518

1.3 0.697

## Soal B4a

5 11.6

10 10.3

15 9.1

20 8.2

25 7.4

30 6.8

## Soal B4b

5 10.5

10 9.2

15 8.2

20 7.4

25 6.7

30 6.1

## Soal B5

1950 33.525

1955 46.519

1960 53.941

1965 72.319

1966 75.160

1967 76.160

1968 84.690

1969 90.866