**FORM LAPORAN AKHIR**

Fandi Ahmad | 2110171033

Metode Gauss-Pivot

Listing Program :

#include <stdio.h>

#include <stdlib.h>

float hasilpers[5];

float peubah[5][5];

float inputPeubah(int x){

int i, j;

for(i=0; i<x; i++){

printf("Persamaan ke-%d \n", i+1);

for(j=0; j<x; j++){

printf("Masukan peubah X%d = ", j+1);

scanf("%f", &peubah[i][j]);

}

printf("Hasil persamaan ke-%d = ", i+1);

scanf("%f", &hasilpers[i]);

printf("\n");

}

}

float getGausPers(int x, int baris, int prevBaris){

baris = baris - 1; // => 2

prevBaris = prevBaris - 1; // => 0

int i, j;

float hasilGaus[5], baru[5], hasilPersBaru;

for(i=0; i<x; i++){

hasilGaus[i] = peubah[prevBaris][i] \* peubah[baris][prevBaris] / peubah[prevBaris][prevBaris];

baru[i] = peubah[baris][i] - hasilGaus[i];

}

hasilPersBaru = hasilpers[prevBaris] \* peubah[baris][prevBaris] / peubah[prevBaris][prevBaris];

hasilpers[baris] -= hasilPersBaru;

for(i=0; i<x; i++){

peubah[baris][i] = baru[i];

}

}

void getHasilPeubah(int z){

float x[5];

int i;

for(i=0; i<z; i++){ x[i] = 0; }

switch(z){

case 2:

x[1] = ( hasilpers[1] - (x[0] \* peubah[1][0]) ) / peubah[1][1];

x[0] = ( hasilpers[0] - (x[1] \* peubah[0][1]) ) / peubah[0][0];

break;

case 3:

x[2] = ( hasilpers[2] - ( ( x[0] \* peubah[2][0] + x[1]\*peubah[2][1] ) ) ) / peubah[2][2] ;

x[1] = ( hasilpers[1] - ( ( x[0] \* peubah[1][0] + x[2]\*peubah[1][2] ) ) ) / peubah[1][1] ;

x[0] = ( hasilpers[0] - ( ( x[1] \* peubah[0][1] + x[2]\*peubah[0][2] ) ) ) / peubah[0][0] ;

break;

case 4:

x[3] = ( hasilpers[3] - ( ( x[0] \* peubah[3][0] + x[1]\*peubah[3][1] + x[2]\*peubah[3][2] ) ) ) / peubah[3][3];

x[2] = ( hasilpers[2] - ( ( x[0] \* peubah[2][0] + x[1]\*peubah[2][1] + x[3]\*peubah[2][3] ) ) ) / peubah[2][2];

x[1] = ( hasilpers[1] - ( ( x[0] \* peubah[1][0] + x[2]\*peubah[1][2] + x[3]\*peubah[1][3] ) ) ) / peubah[1][1];

x[0] = ( hasilpers[0] - ( ( x[1] \* peubah[0][1] + x[2]\*peubah[0][2] + x[3]\*peubah[0][3] ) ) ) / peubah[0][0];

break;

}

for(i=0; i<z; i++){

printf("X%d = %g \n",i, x[i]);

}

}

int main(){

int i, j, pers;

printf("Masukan banyak persamaan = ");

scanf("%d", &pers);

inputPeubah(pers);

for (i=2; i<= pers; i++){

for (j=1; j<i; j++){

getGausPers(pers, i, j);

}

}

printf("=================MATRIKS GAUS===============\n\n");

for(i=0; i<pers; i++){

printf("|\t");

for(j=0; j<pers; j++){

printf("%g\t", peubah[i][j]);

}

printf("| %g\t|", hasilpers[i]);

printf("\n");

}

printf("\n============================================\nHasil : \n");

getHasilPeubah(pers);

}

**CONTOH**

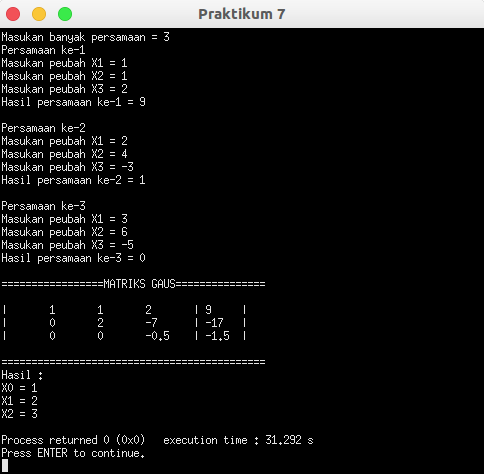
1. Selesaikan sistem linier

x + y + 2z = 9

2x + 4y -3z = 1

3x + 6y -5z = 0

Dengan menggunakan metode Eliminasi Gauss-Pivot serta ketelitian hingga 2 desimal.



1. Selesaikan sistem linier

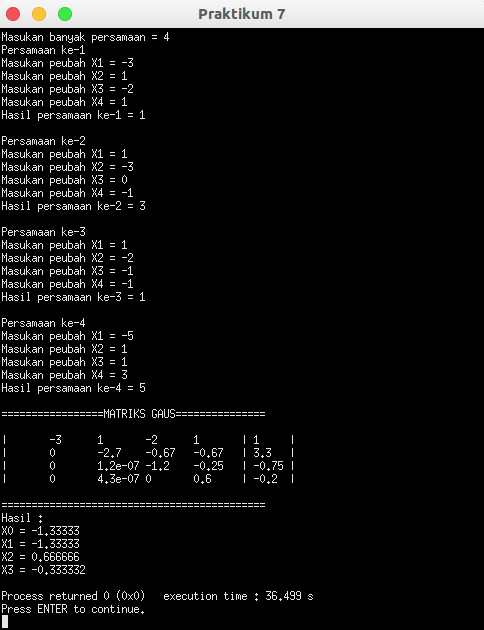
-3x1 + x2 - 2x3 + x4 = 1

x1 – 3x2 – x4 = 3

x1 – 2x2 – x3 – x4 = 1

-5x1 + x2 + x3 + 3x4 = 5

Dengan menggunakan metode eliminasi Gauss-Pivot serta ketelitian hingga 2 desimal.



**LATIHAN**

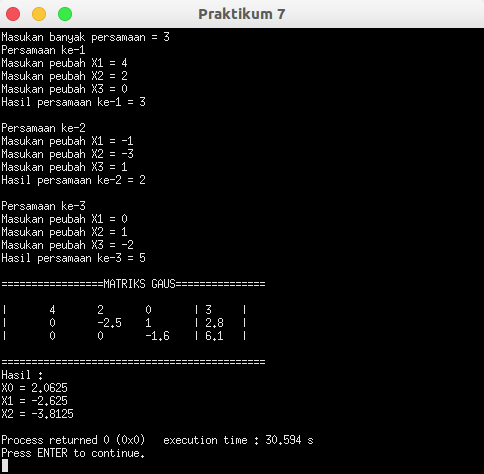
1. Selesaikan sistem linier

4x + 2y = 3

-x – 3y + z = 2

y – 2z = 5

Dengan menggunakan metode eliminasi Gauss-Pivot serta ketelitian hingga 2 desimal.



1. Selesaikan sistem linier

x1 – 2x2 + x4 = 2

2x1 – x2 + 2x3 – 2x4 = 20

-2x1 + 3x2 + 3x3 – 2x4 = 5

4x2 – 3x3 – 3x4 = 8

Dengan menggunakan metode eliminasi Gauss-Pivot serta ketelitian hingga 2 desimal.

