

Nama : Firman Gani Heriansyah

NIM : 21120130043

Penjabaran Matriks Balikan

Link github :

```
#include <iostream>
#include <vector>

using namespace std;

// Membuat matriks identitas
vector<vector<double>> identityMatrix(int n) {
    vector<vector<double>> I(n, vector<double>(n, 0));
    for (int i = 0; i < n; i++)
        I[i][i] = 1;
    return I;
}

// Mencari invers matriks
vector<vector<double>> matrixInverse(vector<vector<double>> A) {
    int n = A.size();
    vector<vector<double>> I = identityMatrix(n);

    for (int i = 0; i < n; i++) {
        double div = A[i][i];
        for (int j = 0; j < n; j++) {
            A[i][j] /= div;
            I[i][j] /= div;
        }
        for (int j = 0; j < n; j++) {
            if (i != j) {
                double factor = A[j][i];
                for (int k = 0; k < n; k++) {
                    A[j][k] -= factor * A[i][k];
                    I[j][k] -= factor * I[i][k];
                }
            }
        }
    }
    return I;
}

// Mencari solusi SPL menggunakan metode invers
vector<double> solveSPL(vector<vector<double>> A, vector<double> b) {
    int n = A.size();
    vector<vector<double>> A_inv = matrixInverse(A);
    vector<double> x(n, 0);
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            x[i] += A_inv[i][j] * b[j];
    return x;
}
```

Fungsi `identityMatrix` digunakan untuk membuat matriks identitas sesuai dengan ukuran matriks masukan. Kemudian, fungsi `matrixInverse` digunakan untuk mencari invers dari matriks koefisien yang diberikan. Proses ini melibatkan operasi pengurangan baris untuk mengubah matriks menjadi bentuk eselon tereduksi dan kemudian mengalikan matriks dengan faktor pembagi untuk mendapatkan matriks invers. Selanjutnya, fungsi `solveSPL` mengambil matriks invers yang telah dihitung sebelumnya untuk mencari solusi SPL dengan mengalikan invers matriks dengan vektor hasil (b). Hasil akhir solusi SPL kemudian dicetak dalam program utama menggunakan matriks koefisien dan vektor hasil yang telah ditentukan.

```
//---Kode Testing---//
int main() {
    // contoh persamaan linear
    // 4x + 12y = 25
    // 5x + 2y = 18
    vector<vector<double>> A = {{4, 12}, {5, 2}};
    vector<double> b = {25, 18};
    vector<double> x = solveSPL(A, b);
    cout << "Firman Gani Heriansyah" << endl;
    cout << "\n" << endl;
    cout << "Solusi SPL:" << endl;
    cout << "x = " << x[0] << endl;
    cout << "y = " << x[1] << endl;
    return 0;
}
```

Kode testing tersebut menyelesaikan SPL dengan didefinisikan dalam bentuk matriks koefisien A dan vektor konstanta b . Sebagai contoh, terdapat dua persamaan linear:

$$4x + 12y = 25$$

$$5x + 2y = 18$$

Kemudian, matriks koefisien A dan vektor konstanta b diinisialisasi menggunakan nilai-nilai dari persamaan tersebut. Setelah itu, fungsi `solveSPL` dipanggil dengan parameter matriks koefisien A

dan vektor konstanta b sebagai argumen. Fungsi `solveSPL` akan mengembalikan solusi SPL dalam bentuk vektor x , di mana setiap elemen vektor x merepresentasikan nilai dari variabel dalam SPL. Setelah solusi SPL diperoleh, program mencetak pesan "Solusi SPL:" diikuti oleh nilai variabel x dan y yang merupakan hasil solusi SPL.

Hasil Output

```
"D:\metnum firman\Matriks E" × + ∨  
Firman Gani Heriansyah  
  
Solusi SPL:  
x = 3.19231  
y = 1.01923  
  
Process returned 0 (0x0)   execution time : 0.045 s  
Press any key to continue.  
|
```

Link github: <https://github.com/Frmngh/Metodenumerik>