

LAPORAN PRAKTIKUM

JOBSCHEET 14 Matriks



Nama Anggota Kelompok :

Andy Otani Dipo Yudho / 244107020241

Nadya Aurora Gebi Agista / 244107020034

Singgih Wahyu Permana / 244107020102 (PL)

Kelas : TI 1H

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

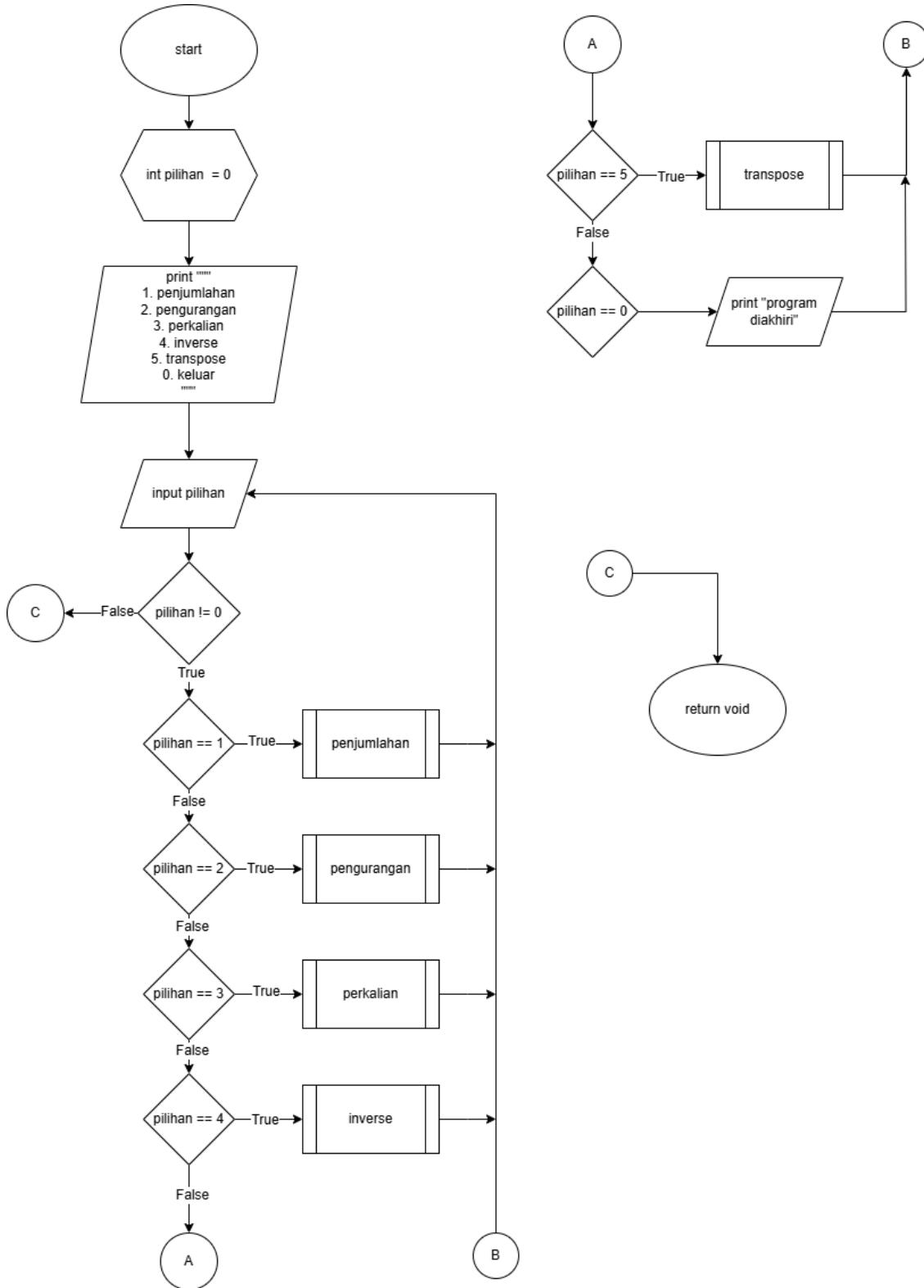
JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

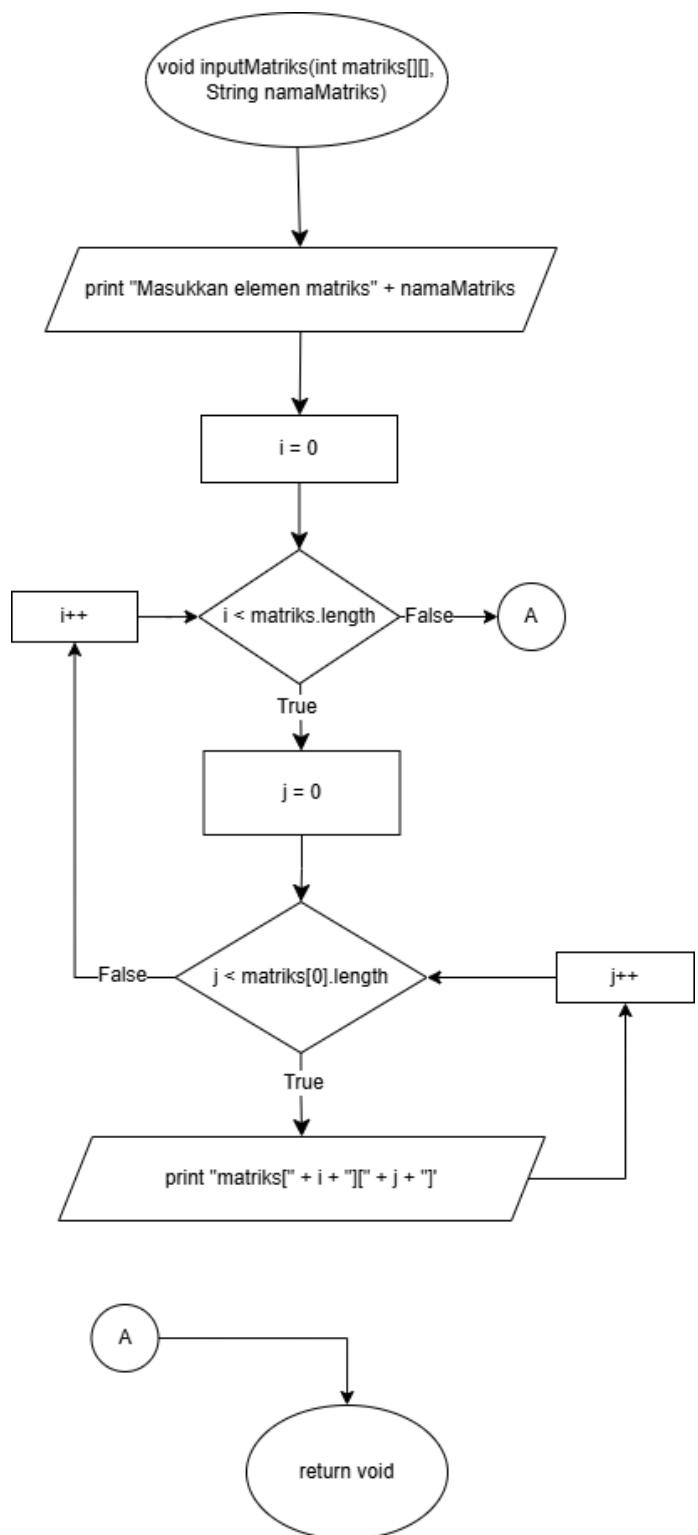
2024

FLOWCHART

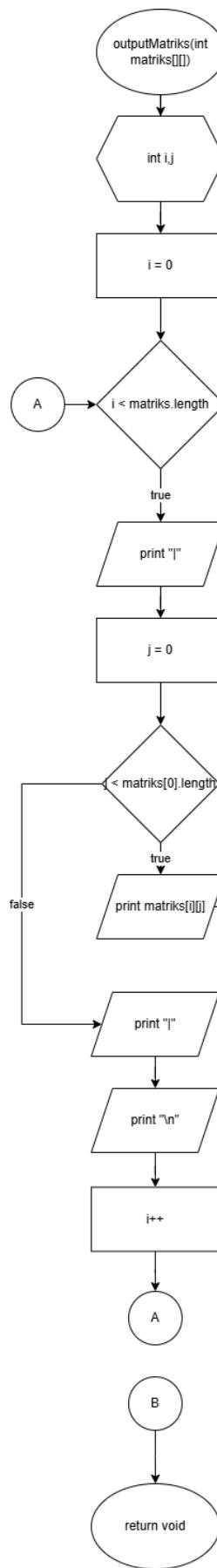
- Fungsi main()



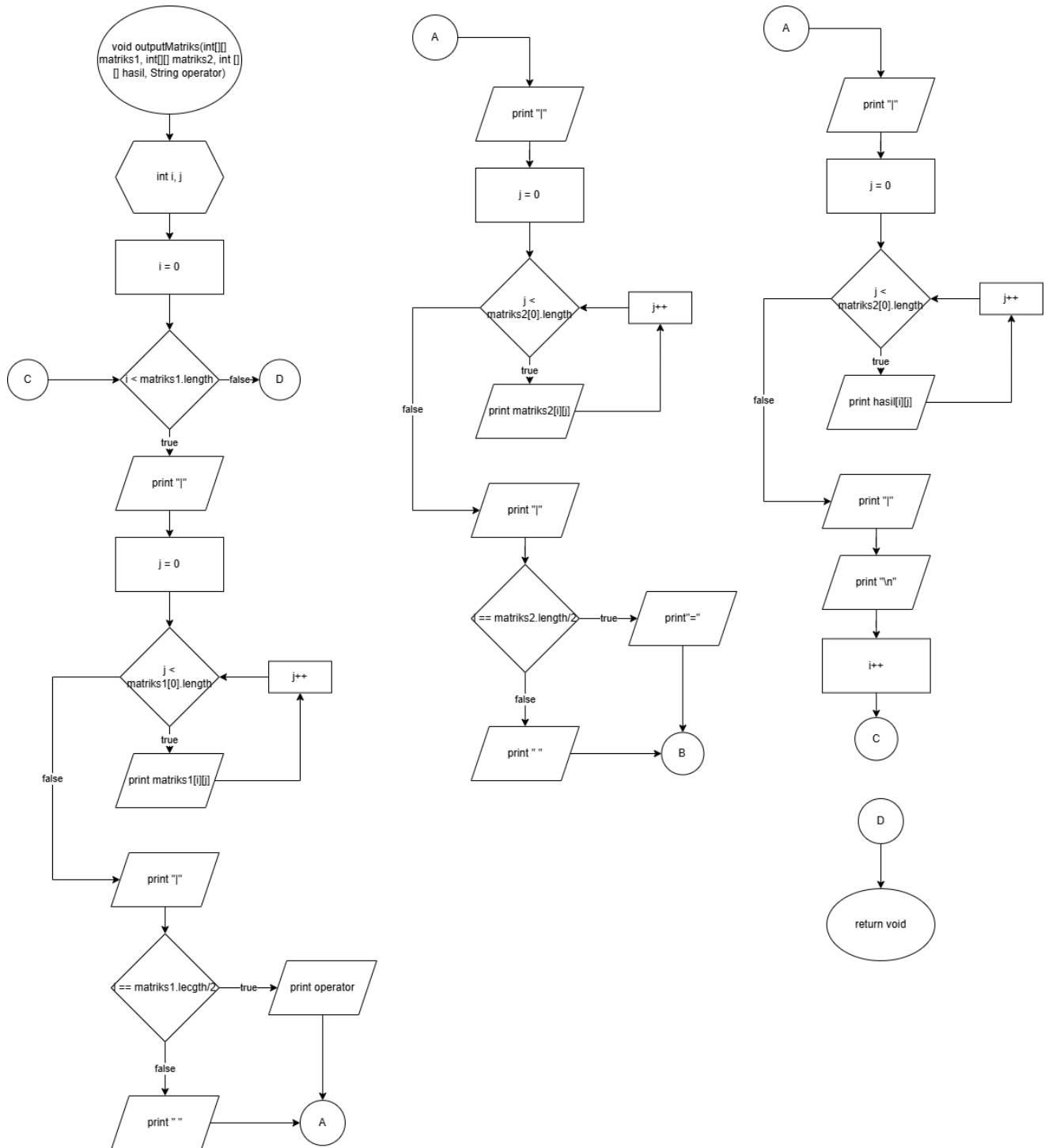
- Fungsi void inputMatriks(int matriks[], String namaMatriks)



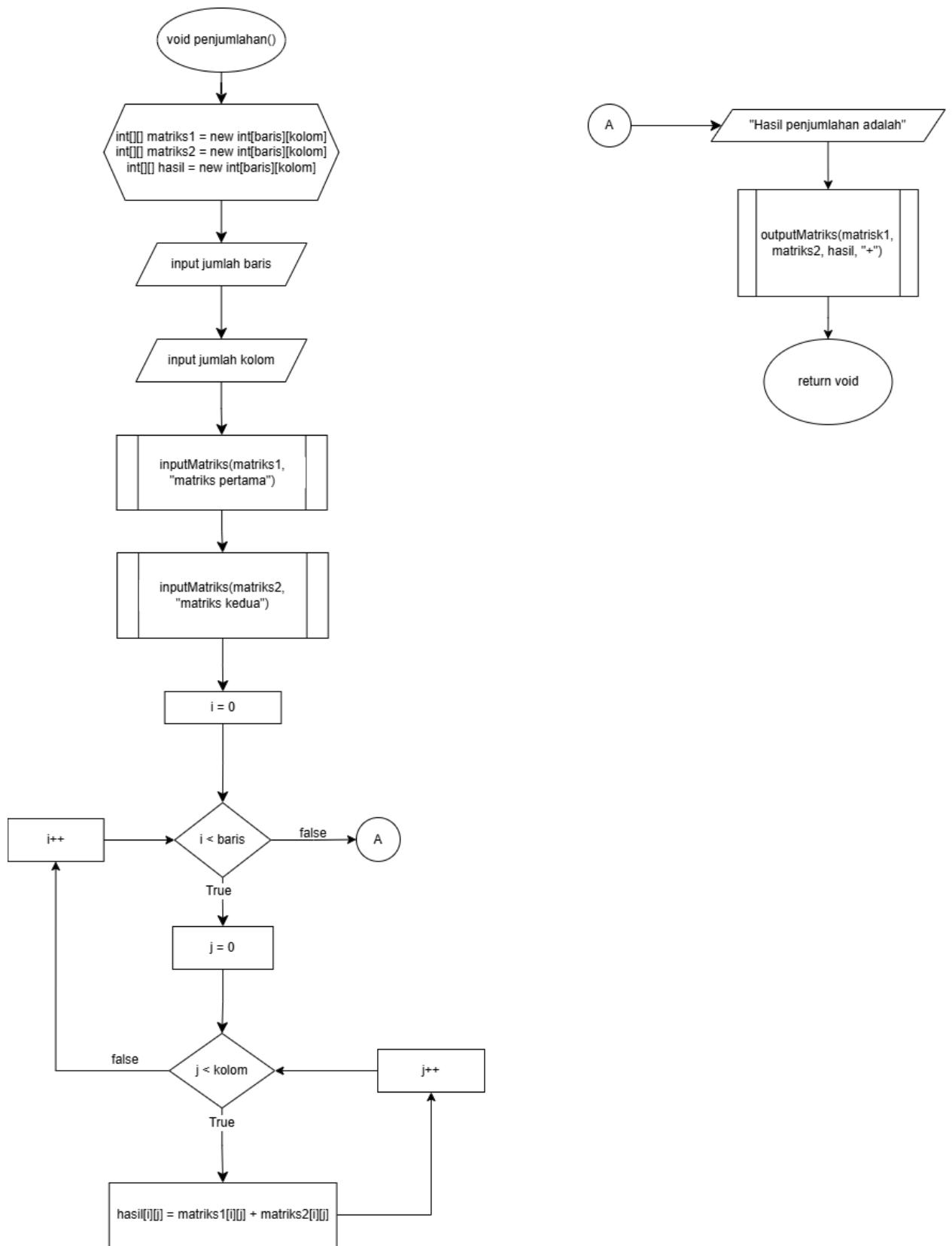
- Fungsi outputMatriks(int matriks[][])



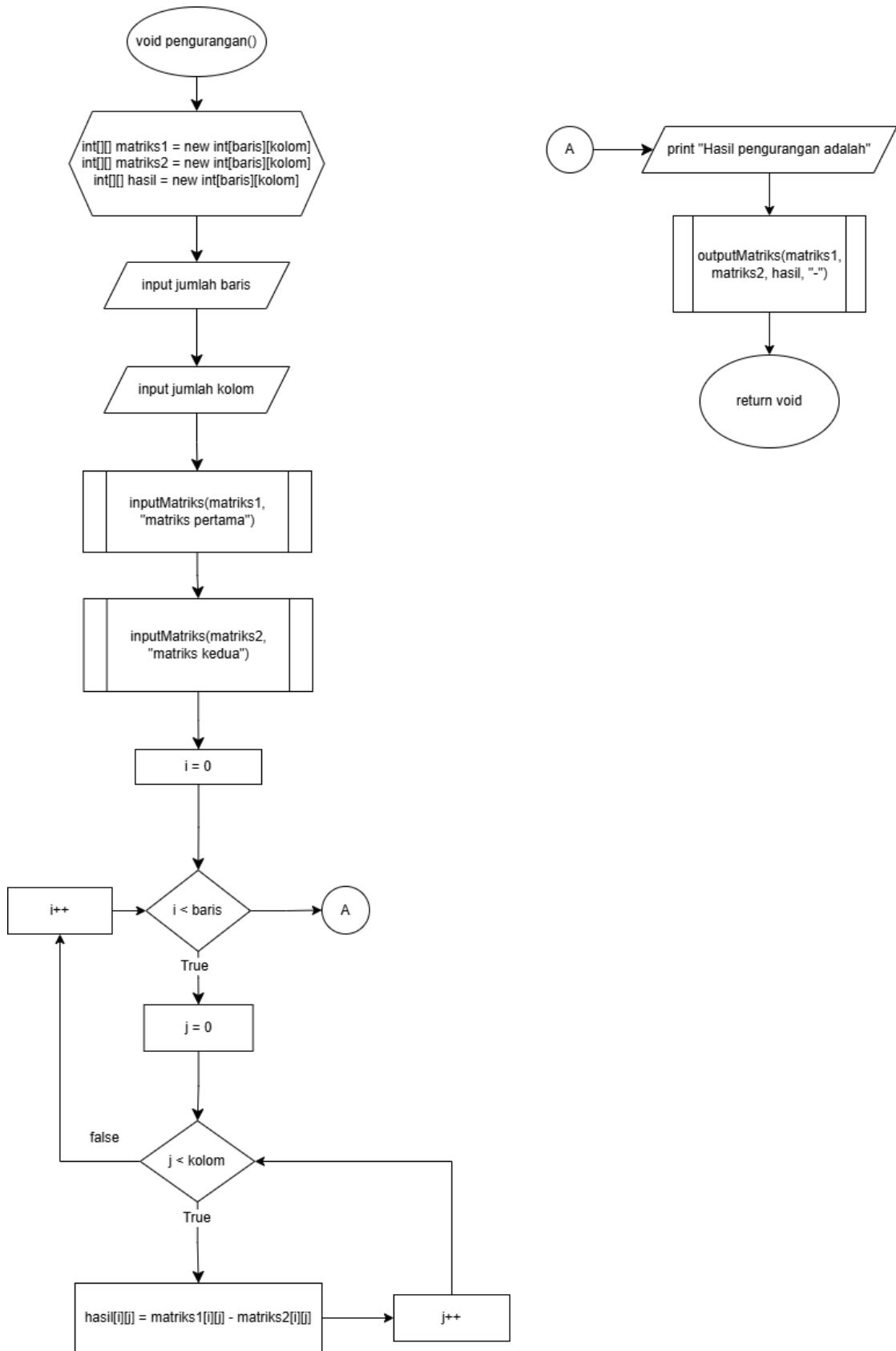
- Fungsi void outputMatriks(int[][] matriks1, int[][] matriks2, int [][] hasil, String operator)



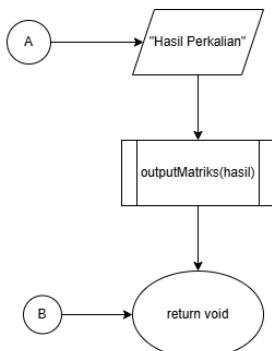
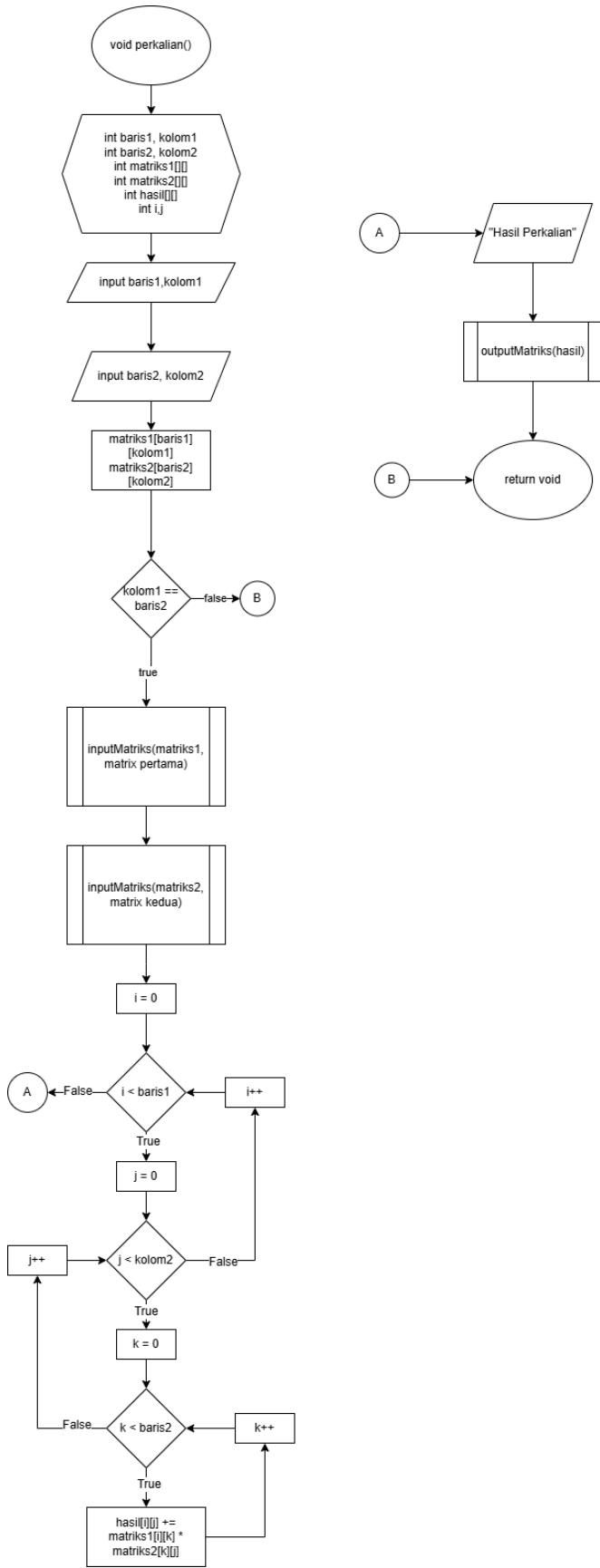
- Fungsi penjumlahan()



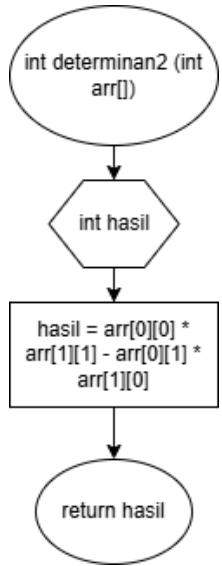
- Fungsi pengurangan()



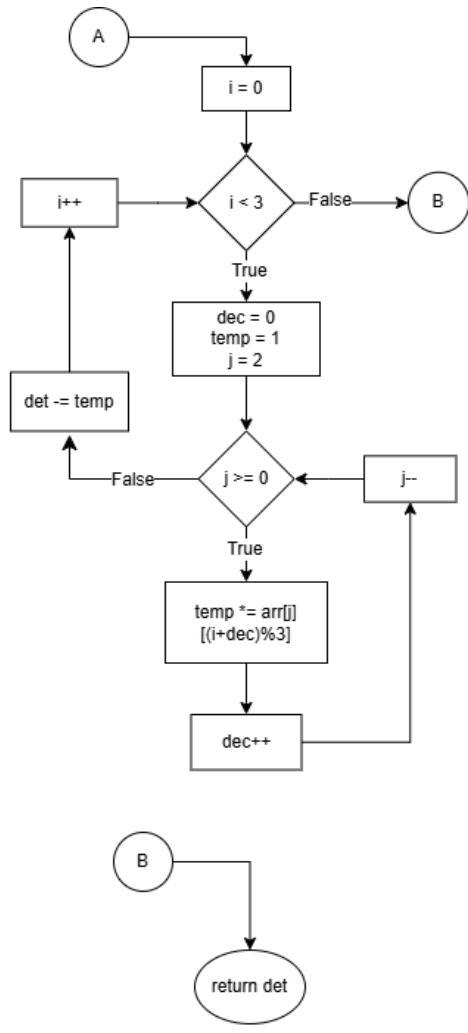
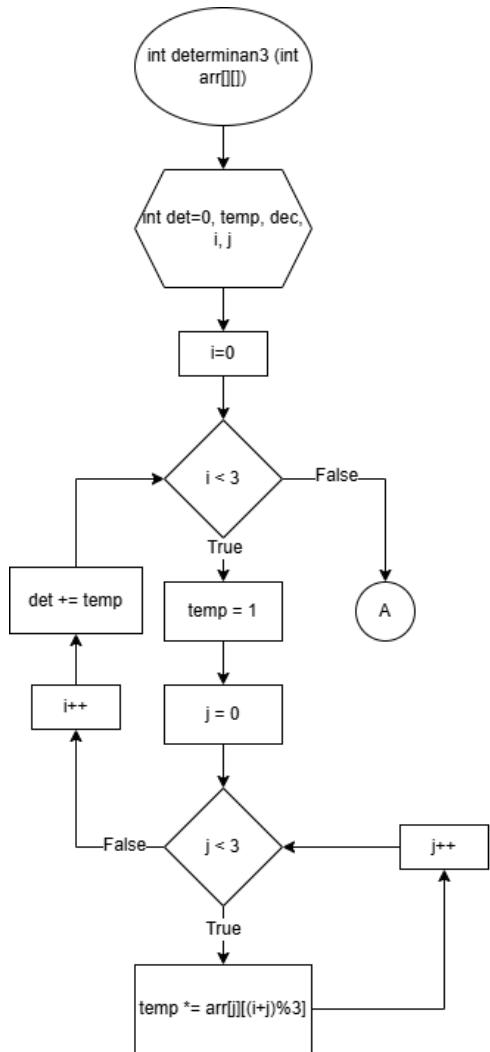
- Fungsi perkalian()



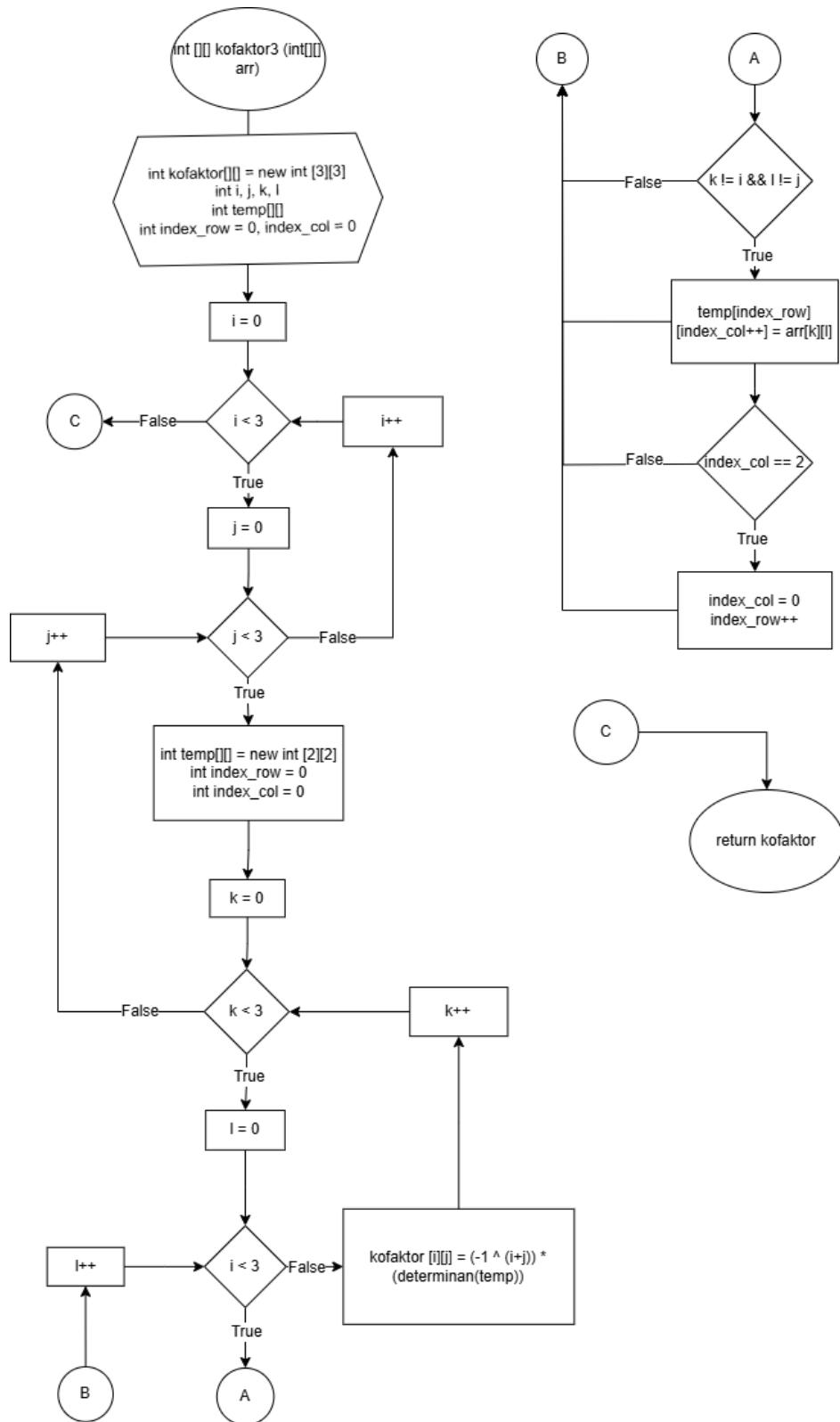
- determinan2 (int arr[])



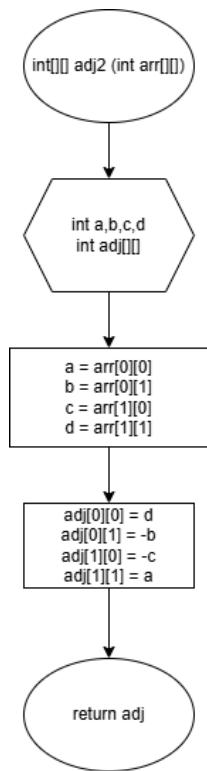
- determinan3 (int arr[])



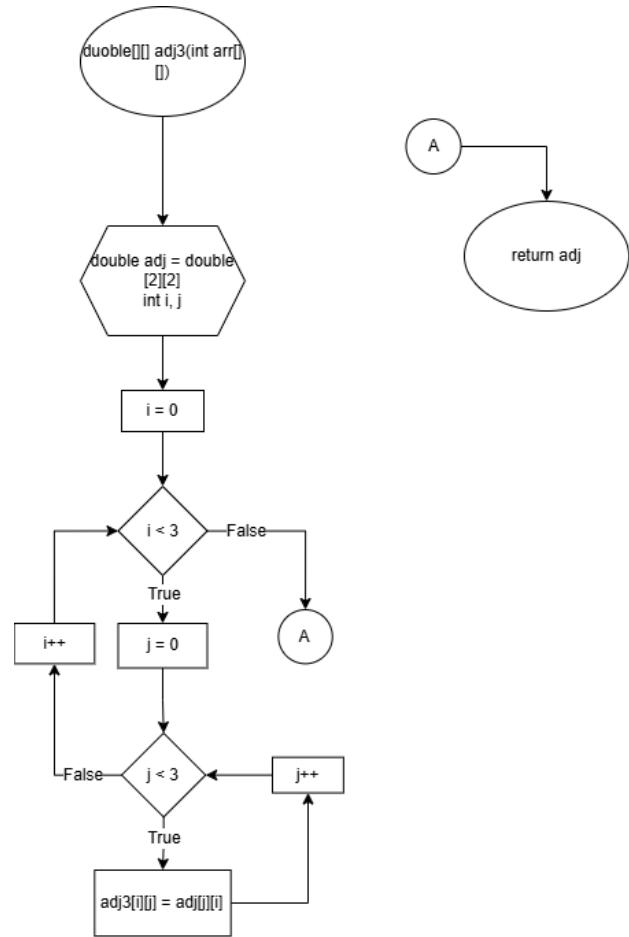
- kofaktor3 (int[][]arr)



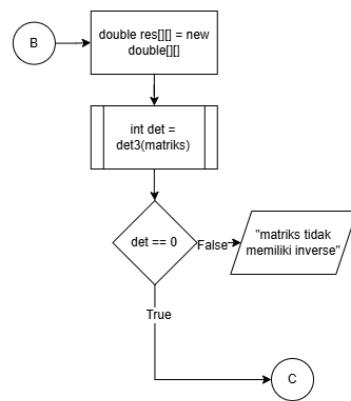
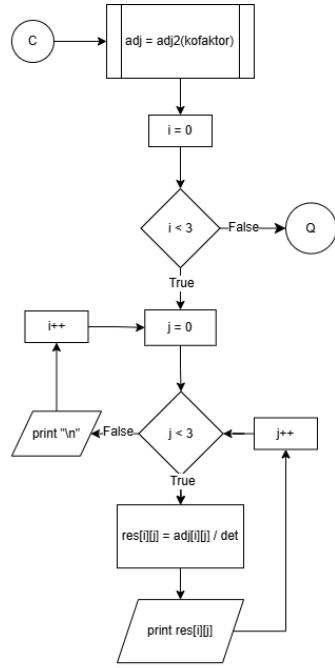
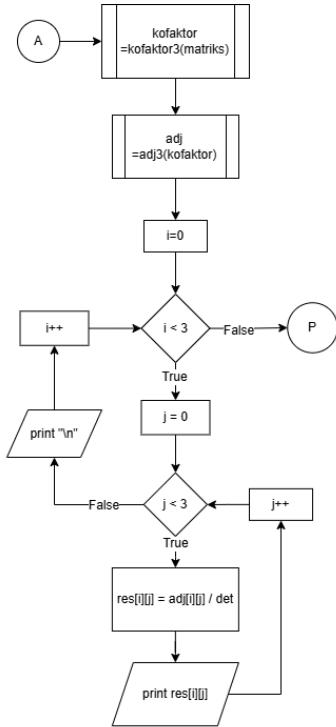
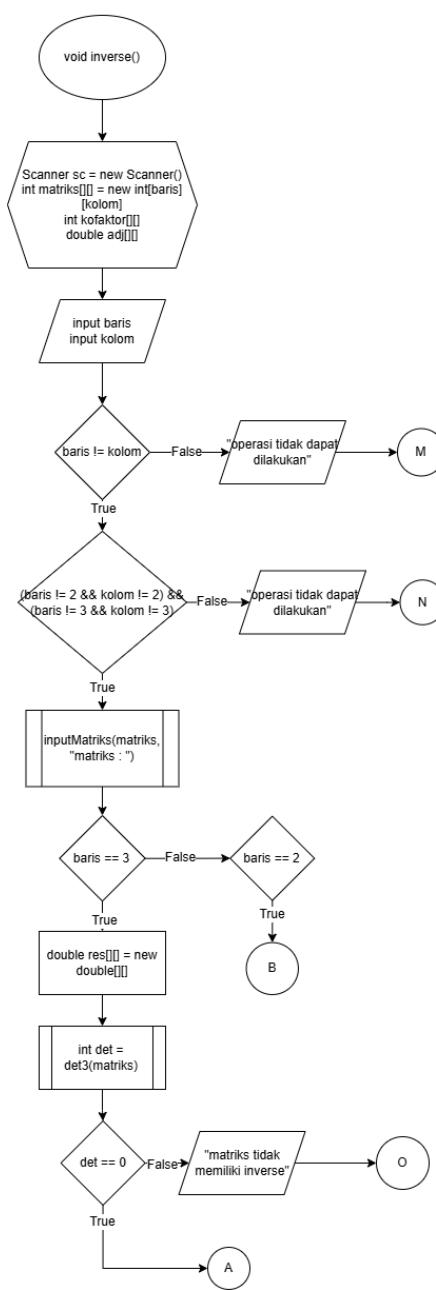
- int[][] adj2 (int arr[][])



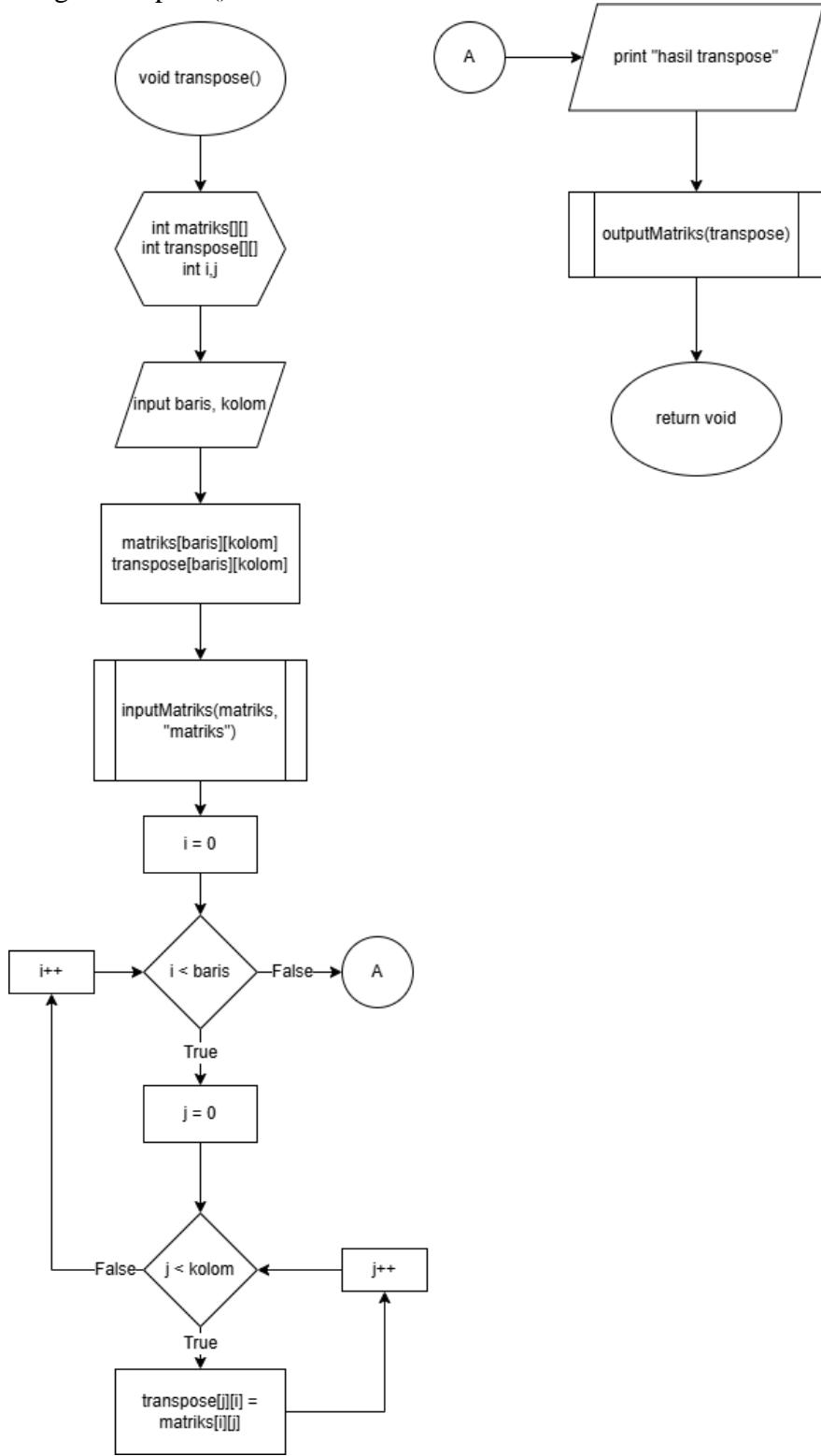
- double[][] adj3(int arr[][])



- Fungsi inverse()



- Fungsi transpose()



KODE PROGRAM

```
import java.util.Scanner;

public class Matriks {
    public static void inputMatriks(int matriks[][], String namaMatriks) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Masukan elemen matriks " + namaMatriks);
        for (int i = 0; i < matriks.length; i++) {
            for (int j = 0; j < matriks[0].length; j++) {
                System.out.print("matriks[" + i + "][" + j + "]: ");
                matriks[i][j] = sc.nextInt();
            }
        }
    }

    public static void outputMatriks(int matriks[][]) {
        for (int i = 0; i < matriks.length; i++) {
            System.out.print("| ");
            for (int j = 0; j < matriks[0].length; j++) {
                System.out.print(matriks[i][j] + " ");
            }
            System.out.print(" | ");
            System.out.println();
        }
    }

    public static void outputMatriks(int matriks[][], int matriks2[][], int matriks3[][], String operator) {
        for(int i=0;i<matriks.length;i++){
            System.out.print(" | ");
            for(int j=0;j<matriks[0].length;j++){
                System.out.print(" " + matriks[i][j] + " ");
            }
            System.out.print(" | ");
            if(i==matriks.length/2){
                System.out.print(" " + operator + " ");
            }else{
                System.out.print("      ");
            }
            System.out.print(" | ");
            for(int j=0;j<matriks2[0].length;j++){
                System.out.print(" " + matriks2[i][j] + " ");
            }
            System.out.print(" | ");
            if(i==matriks.length/2){
                System.out.print(" = ");
            }else{
                System.out.print("      ");
            }
        }
    }
}
```

```
        System.out.print("    ");
    }
    System.out.print(" | ");
    for(int j=0;j<matriks2[0].length;j++) {
        System.out.print(" " + matriks3[i][j] + " ");
    }
    System.out.print(" | ");

    System.out.println();
}
}

public static void penjumlahan() {
    Scanner sc = new Scanner(System.in);

    System.out.print("Masukkan jumlah baris matriks: ");
    int baris = sc.nextInt();
    System.out.print("Masukkan jumlah kolom matriks: ");
    int kolom = sc.nextInt();

    int[][] matriks1 = new int[baris][kolom];
    int[][] matriks2 = new int[baris][kolom];
    int[][] hasil = new int[baris][kolom];

    inputMatriks(matriks1, "matriks pertama : ");
    inputMatriks(matriks2, "matriks kedua : ");

    for (int i = 0; i < baris; i++) {
        for (int j = 0; j < kolom; j++) {
            hasil[i][j] = matriks1[i][j] + matriks2[i][j];
        }
    }

    System.out.println("Hasil penjumlahan matriks:");
    outputMatriks(matriks1, matriks2, hasil, "+");
}

public static void pengurangan() {
    Scanner sc = new Scanner(System.in);

    System.out.print("Masukkan jumlah baris matriks: ");
    int baris = sc.nextInt();
    System.out.print("Masukkan jumlah kolom matriks: ");
    int kolom = sc.nextInt();

    int[][] matriks1 = new int[baris][kolom];
    int[][] matriks2 = new int[baris][kolom];
    int[][] hasil = new int[baris][kolom];

    inputMatriks(matriks1, "matriks pertama : ");
    inputMatriks(matriks2, "matriks kedua : ");
```

```
        for (int i = 0; i < baris; i++) {
            for (int j = 0; j < kolom; j++) {
                hasil[i][j] = matriks1[i][j] - matriks2[i][j];
            }
        }
        System.out.println("Hasil pengurangan adalah ");
        outputMatriks(matriks1, matriks2, hasil, "-");
    }

    public static void perkalian() {
        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan jumlah baris matriks 1: ");
        int baris1 = sc.nextInt();
        System.out.print("Masukkan jumlah kolom matriks 1: ");
        int kolom1 = sc.nextInt();

        System.out.print("Masukkan jumlah baris matriks 2: ");
        int baris2 = sc.nextInt();
        System.out.print("Masukkan jumlah kolom matriks 2: ");
        int kolom2 = sc.nextInt();

        if (kolom1 != baris2) {
            System.out.println("Perkalian tidak dapat dilakukan!");
            return;
        }

        int[][] matriks1 = new int[baris1][kolom1];
        int[][] matriks2 = new int[baris2][kolom2];
        int[][] hasil = new int[baris1][kolom2];

        inputMatriks(matriks1, "matriks pertama : ");
        inputMatriks(matriks2, "matriks kedua : ");

        for (int i = 0; i < baris1; i++) {
            for (int j = 0; j < kolom2; j++) {
                hasil[i][j] = 0;
                for (int k = 0; k < kolom1; k++) {
                    hasil[i][j] += matriks1[i][k] * matriks2[k][j];
                }
            }
        }
        System.out.println("Hasil perkalian adalah ");
        outputMatriks(hasil);
    }

    public static int determinan2(int arr[][])
    {
        return arr[0][0] * arr[1][1] - arr[0][1] * arr[1][0];
    }
}
```

```

public static int determinan3(int arr[][]) {
    int det = 0;
    for (int i = 0; i < 3; i++) {
        int temp = 1;
        for (int j = 0; j < 3; j++) {
            temp *= arr[j][(i + j) % 3];
        }
        det += temp;
    }

    for (int i = 0; i < 3; i++) {
        int dec = 0;
        int temp = 1;
        for (int j = 3 - 1; j >= 0; j--) {
            temp *= arr[j][(i + dec) % 3];
            dec++;
        }
        det -= temp;
    }

    return det;
}

public static int[][] kofaktor3(int arr[][]) {
    int kofaktor[][] = new int[3][3];
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            int[][] temp = new int[2][2];
            int indexRow = 0, indexCol = 0;
            for (int k = 0; k < 3; k++) {
                for (int l = 0; l < 3; l++) {
                    if (k != i && l != j) {
                        temp[indexRow][indexCol++] = arr[k][l];
                        if (indexCol == 2) {
                            indexCol = 0;
                            indexRow++;
                        }
                    }
                }
            }
            kofaktor[i][j] = (int) Math.pow(-1, i + j) * determinan2(temp);
        }
    }
    return kofaktor;
}

public static double[][] getAdjoint2(int arr[][]) {
    int a = arr[0][0];
    int b = arr[0][1];
    int c = arr[1][0];
    int d = arr[1][1];
}

```

```
        double adj[][] = new double[2][2];
        adj[0][0] = d;
        adj[0][1] = -b;
        adj[1][0] = -c;
        adj[1][1] = a;
        return adj;
    }

    public static double[][] getAdjoint3(int arr[][][]) {
        double adj[][] = new double[3][3];
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                adj[i][j] = arr[j][i];
            }
        }
        return adj;
    }

    public static void inverse() {
        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan jumlah baris: ");
        int baris = sc.nextInt();
        System.out.print("Masukkan jumlah kolom: ");
        int kolom = sc.nextInt();

        if (baris != kolom) {
            System.out.println("Matriks tidak memiliki inverse");
            return;
        }

        if ((baris != 2 && kolom != 2) && (baris != 3 && kolom != 3)) {
            System.out.println("Operasi tidak dapat dilakukan");
            return;
        }

        int[][] matriks = new int[baris][kolom];

        inputMatriks(matriks, "matriks : ");

        if (baris == 3) {
            double res[][] = new double[3][3];
            int det = determinan3(matriks);
            if (det == 0) {
                System.out.println("Matriks tidak memiliki inverse");
                return;
            }
            int kofaktor[][] = kofaktor3(matriks);
            double adj[][] = getAdjoint3(kofaktor);
            matriks = adj;
        }
    }
}
```

```
        System.out.println("Hasil Inverse Matriks: ");
        for (int i = 0; i < 3; i++) {
            System.out.print(" | ");
            for (int j = 0; j < 3; j++) {
                res[i][j] = adj[i][j] / det;
                System.out.printf("% .2f ", res[i][j]);
            }
            System.out.print(" | ");
            System.out.println();
        }
    } else if (baris == 2) {
        double res[][] = new double[2][2];
        int det = determinan2(matriks);
        if (det == 0) {
            System.out.println("Matriks tidak memiliki inverse");
            return;
        }
        double adj[][] = getAdjoint2(matriks);
        System.out.println("Hasil Inverse Matriks: ");
        for (int i = 0; i < 2; i++) {
            for (int j = 0; j < 2; j++) {
                res[i][j] = adj[i][j] / det;
                System.out.printf("% .2f ", (float) res[i][j]);
            }
            System.out.println();
        }
    }
}

public static void transpose() {
    Scanner sc = new Scanner(System.in);

    System.out.print("Masukkan jumlah baris: ");
    int baris = sc.nextInt();
    System.out.print("Masukkan jumlah kolom: ");
    int kolom = sc.nextInt();

    int[][] matriks = new int[baris][kolom];
    int[][] transpose = new int[kolom][baris];

    inputMatriks(matriks, "matriks");

    for (int i = 0; i < baris; i++) {
        for (int j = 0; j < kolom; j++) {
            transpose[j][i] = matriks[i][j];
        }
    }
    System.out.println("Hasil transpose adalah ");
    outputMatriks(transpose);
}
```

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int pilihan = 0;
    do {
        System.out.println("===== MENU =====");
        System.out.println("1. Penjumlahan");
        System.out.println("2. Pengurangan");
        System.out.println("3. Perkalian");
        System.out.println("4. Inverse");
        System.out.println("5. Transpose");
        System.out.println("0. Keluar");
        System.out.print("Pilihan: ");
        pilihan = sc.nextInt();
        switch (pilihan) {
            case 1:
                penjumlahan();
                break;
            case 2:
                pengurangan();
                break;
            case 3:
                perkalian();
                break;
            case 4:
                inverse();
                break;
            case 5:
                transpose();
                break;
            case 0:
                System.out.println("Program selesai");
                return;
            default:
                System.out.println("Pilihan tidak tersedia");
                break;
        }
    } while (pilihan!=0);
}
```

HASIL OUTPUT :

- Pilihan 1 dan 2 (Penjumlahan dan Pengurangan)

```
=====
=           MENU          =
=====
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Inverse
5. Transpose
0. Keluar
Pilihan: 1
Masukkan jumlah baris matriks: 2
Masukkan jumlah kolom matriks: 2
Masukan elemen matriks matriks pertama :
matriks[0][0]: 5
matriks[0][1]: 6
matriks[1][0]: 3
matriks[1][1]: 1
Masukan elemen matriks matriks kedua :
matriks[0][0]: 2
matriks[0][1]: 4
matriks[1][0]: 8
matriks[1][1]: 3
Hasil penjumlahan matriks:
| 5 6 | + | 2 4 | = | 7 10 |
| 3 1 |     | 8 3 |     | 11 4 |
=====

=====
=           MENU          =
=====
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Inverse
5. Transpose
0. Keluar
Pilihan: 2
Masukkan jumlah baris matriks: 3
Masukkan jumlah kolom matriks: 3
Masukan elemen matriks matriks pertama :
matriks[0][0]: 10
matriks[0][1]: 28
matriks[0][2]: 15
matriks[1][0]: 16
matriks[1][1]: 13
matriks[1][2]: 13
matriks[2][0]: 24
matriks[2][1]: 27
matriks[2][2]: 20
Masukan elemen matriks matriks kedua :
matriks[0][0]: 9
matriks[0][1]: 22
matriks[0][2]: 10
matriks[1][0]: 14
matriks[1][1]: 10
matriks[1][2]: 5
matriks[2][0]: 20
matriks[2][1]: 19
matriks[2][2]: 8
Hasil pengurangan adalah
| 10 28 15 | - | 9 22 10 | = | 1 6 5 |
| 16 13 13 |     | 14 10 5 |     | 2 3 8 |
| 24 27 20 |     | 20 19 8 |     | 4 8 12 |
```

- Pilihan 3 dan 4 (Perkalian dan Inverse)

```
=====
=           MENU          =
=====

1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Inverse
5. Transpose
0. Keluar
Pilihan: 3
Masukkan jumlah baris matriks 1: 2
Masukkan jumlah kolom matriks 1: 2
Masukkan jumlah baris matriks 2: 2
Masukkan jumlah kolom matriks 2: 3
Masukan elemen matriks matriks pertama :
matriks[0][0]: 4
matriks[0][1]: 2
matriks[1][0]: 1
matriks[1][1]: 7
Masukan elemen matriks matriks kedua :
matriks[0][0]: 5
matriks[0][1]: 3
matriks[0][2]: 4
matriks[1][0]: 8
matriks[1][1]: 6
matriks[1][2]: 1
Hasil perkalian adalah
| 36 24 18 |
| 61 45 11 |
=====
=           MENU          =
=====

1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Inverse
5. Transpose
0. Keluar
Pilihan: 4
Masukkan jumlah baris: 3
Masukkan jumlah kolom: 3
Masukan elemen matriks matriks :
matriks[0][0]: 2
matriks[0][1]: 4
matriks[0][2]: 5
matriks[1][0]: 3
matriks[1][1]: 5
matriks[1][2]: 6
matriks[2][0]: 1
matriks[2][1]: 9
matriks[2][2]: 8
Hasil Inverse Matriks:
| -1.40 1.30 -0.10 |
| -1.80 1.10 0.30 |
| 2.20 -1.40 -0.20 |
```

- Pilihan 5 dan 0 (Transpose dan Keluar)

```
=====
=           MENU          =
=====

1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Inverse
5. Transpose
0. Keluar
Pilihan: 5
Masukkan jumlah baris: 2
Masukkan jumlah kolom: 3
Masukan elemen matriks matriks
matriks[0][0]: 1
matriks[0][1]: 2
matriks[0][2]: 3
matriks[1][0]: 4
matriks[1][1]: 5
matriks[1][2]: 6
Hasil transpose adalah
| 1 4 |
| 2 5 |
| 3 6 |
=====
=           MENU          =
=====

1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Inverse
5. Transpose
0. Keluar
Pilihan: 0
Program selesai
```