

1.รายละเอียดโครงการ

1.1 ชื่อโครงการ

โปรแกรมเครื่องคิดเลขเมทริกซ์ Matrix Calculator

1.2 สมาชิกกลุ่ม

1.นาย กิตติคุณ สุวรรณศิริ ภาควิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยกรุงเทพ

2.นาย ภูมิพัฒน์ รัตนยืนยง ภาควิชาวิศวกรรมคอมพิวเตอร์
มหาวิทยาลัยกรุงเทพ

1.3 อาจารย์ที่ปรึกษาโครงการ

อาจารย์

ภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยกรุงเทพ

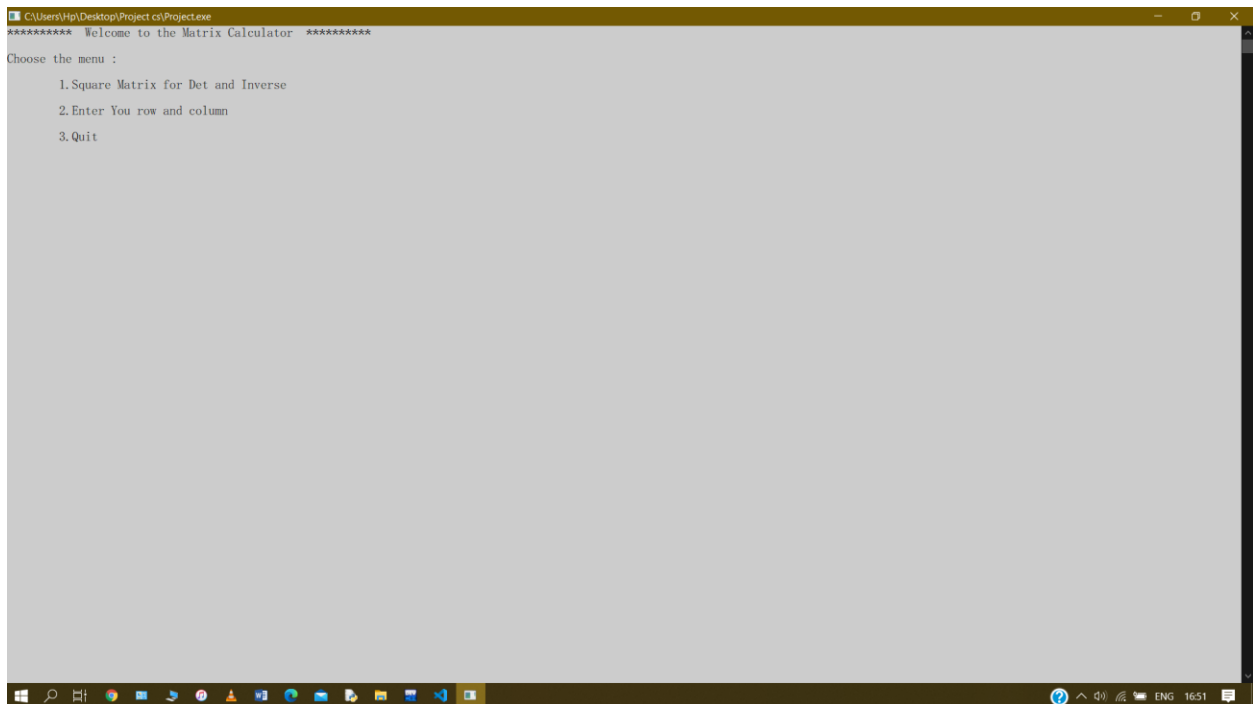
2 สารสำคัญและคำสำคัญของโครงการ

3. หลักการและเหตุผล

4.วัตถุประสงค์ของโครงการ

7. รายละเอียดของการพัฒนา

การทำงานของโปรแกรม



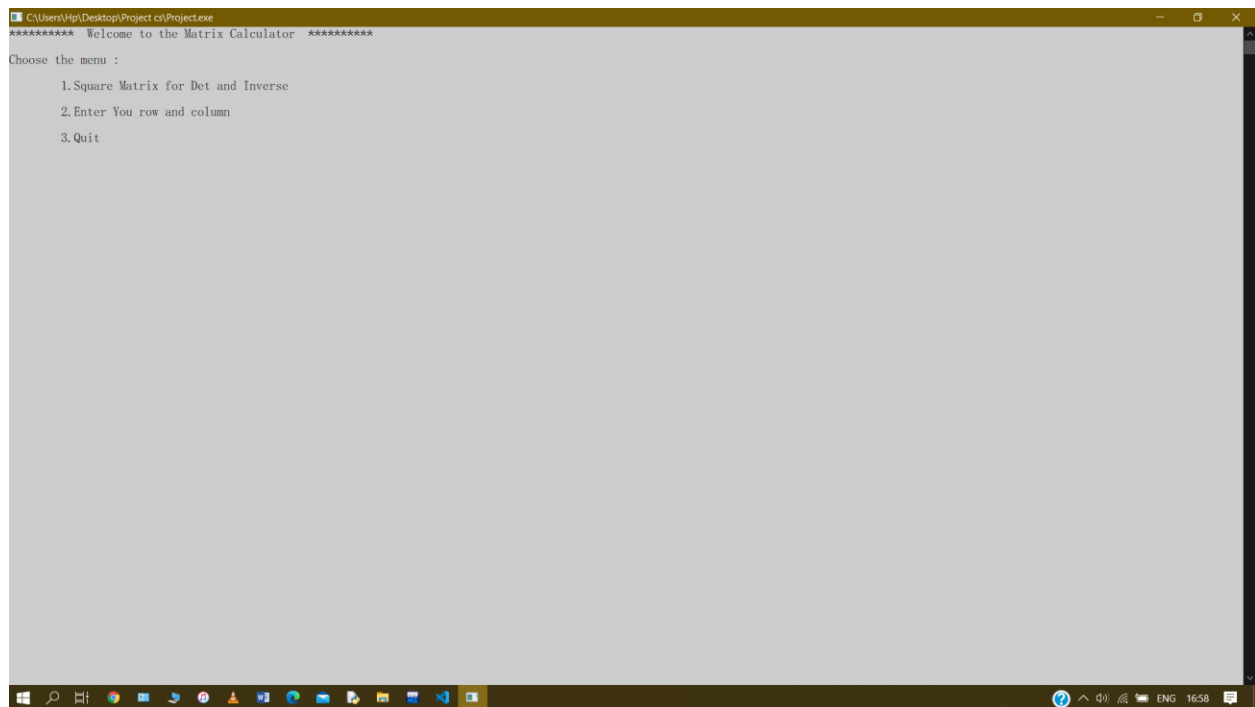
เมื่อเปิดโปรแกรมขึ้นมา

โปรแกรมจะให้เลือกเมนู

1. เมทริกซ์จัตุรัส

2. เมทริกซ์แบบที่เราสามารถกำหนดแถวและคอลัมน์ได้เอง

3. ออกจากโปรแกรม



โดยเมื่อเลือกเมนูที่ 1

โปรแกรมจะถามว่าต้องสร้างเมทริกซ์จัตุรัสกี่มิติ

โดยเราสามารถกำหนดได้เอง ในตัวอย่างกำหนด 2 มิติ

เมื่อกำหนดมิติเสร็จแล้ว โปรแกรมจะให้เราได้ตัวเลขลงไป

```
C:\Users\Hp\Desktop\Project cs\Project.exe
***** Welcome to the Matrix Calculator *****

Choose the menu :

    1. Square Matrix for Det and Inverse
    2. Enter You row and column
    3. Quit

1
Enter size of a Square Matrix (Must more than 1) :
2
Enter the number of 2 x 2 matrix:
2
4
6
8

The matrix is
2.00  4.00
6.00  8.00

Do you want to

    1. Determinant
    2. Inverse
```

เมื่อใส่เลขครบแล้ว

โปรแกรมจะแสดงค่าที่เราใส่ลงไปและถามว่าต้องการจะทำอะไร
โดยมีตัวเลือกว่า

1. หา Determinant
2. หา Inverse

```
C:\Users\Hp\Desktop\Project\cs\Project.exe
***** Welcome to the Matrix Calculator *****

Choose the menu :

    1. Square Matrix for Det and Inverse
    2. Enter You row and column
    3. Quit

1
Enter size of a Square Matrix (Must more than 1) :
2
Enter the number of 2 x 2 matrix:
2
4
6
8
The matrix is
2.00  4.00
6.00  8.00
Do you want to
    1. Determinant
    2. Inverse
1
Det of matrix is :
-8.00
Do you want to continue using the program? .(Y/N) :
-
```

ในที่นี้เลือกเมนู Determinant
โปรแกรมจะคำนวณค่าจากตัวเลขที่เราใส่ลงไป
หลังจากนั้น โปรแกรมจะถามว่าต้องการใช้งานต่อหรือไม่

```
C:\Users\Hp\Desktop\Project\cs\Project.exe
***** Welcome to the Matrix Calculator *****

Choose the menu :

    1. Square Matrix for Det and Inverse
    2. Enter You row and column
    3. Quit
```

เมื่อตอบ y หรือ Y โปรแกรมจะล้างหน้าจอและแสดงเมนูหลักอีกครั้ง

```
C:\Users\Hp\Desktop\Project>cs\Project.exe
***** Welcome to the Matrix Calculator *****

Choose the menu :

1. Square Matrix for Det and Inverse
2. Enter You row and column
3. Quit

2

Enter the number of rows (between 1 and 100): 2
Enter the number of columns (between 1 and 100): 3
Row and Column are 2 3
Enter number of 1st matrix:
Enter number a [0][0]:
```

เมื่อเลือกเมนูที่ 2 โปรแกรมจะให้เรากำหนดว่าเราต้องการสร้างเมทริกซ์กี่แถวและกี่คอลัมน์ระหว่าง 1 ถึง 100
ในที่นี้เลือก 2 แถว 3 คอลัมน์
จากนั้นโปรแกรมจะให้เราได้ตัวเลขที่เราต้องการลงไปเมทริกซ์ 1 และเมทริกซ์ 2

```
C:\Users\Hp\Desktop\Project cs\Project.exe
3. Quit
2
Enter the number of rows (between 1 and 100): 2
Enter the number of columns (between 1 and 100): 3
Row and Column are 2 3
Enter number of 1st matrix:
Enter number a [0][0]: 1
Enter number a [0][1]: 2
Enter number a [0][2]: 3
Enter number a [1][0]: 4
Enter number a [1][1]: 5
Enter number a [1][2]: 6
Enter number of 2nd matrix:
Enter number b [0][0]: 9
Enter number b [0][1]: 8
Enter number b [0][2]: 7
Enter number b [1][0]: 6
Enter number b [1][1]: 5
Enter number b [1][2]: 4
Entered matrix 1st:
1.00  2.00  3.00
4.00  5.00  6.00
Entered matrix 2nd:
9.00  8.00  7.00
6.00  5.00  4.00
Do you want to
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Determinant (Row and Column must be Equal and Must more than 1)
6. Inverse (Row and Column must be Equal and Must more than 1)
```

เมื่อใส่ตัวเลขครบแล้ว โปรแกรมจะแสดงตัวเลขที่เราใส่ลงไป
จากนั้นโปรแกรมจะถามต่อว่าเราต้องการจะทำอะไรกับเมทริกซ์ โดย

1. นำเมทริกซ์ทั้ง 2 มาบวกกัน
2. นำเมทริกซ์ทั้ง 2 มาลบกัน
3. นำเมทริกซ์ทั้ง 2 มาคูณกัน
4. นำเมทริกซ์ทั้ง 2 มาหารกัน
5. หา Determinant ของเมทริกซ์ที่ 1 หรือ 2
โดยแถวและคอลัมน์ต้องเท่ากันและมากกว่า 1
6. หา Inverse ของเมทริกซ์ที่ 1 หรือ 2
โดยแถวและคอลัมน์ต้องเท่ากันและมากกว่า 1

```
C:\Users\Hp\Desktop\Project cs\Project.exe
Enter number a [1][1]: 5
Enter number a [1][2]: 6
Enter number of 2nd matrix:
Enter number b [0][0]: 9
Enter number b [0][1]: 8
Enter number b [0][2]: 7
Enter number b [1][0]: 6
Enter number b [1][1]: 5
Enter number b [1][2]: 4

Entered matrix 1st:
1.00  2.00  3.00
4.00  5.00  6.00
Entered matrix 2nd:
9.00  8.00  7.00
6.00  5.00  4.00

Do you want to
    1.Addition
    2.Subtraction
    3.Multiplication
    4.Division
    5.Determinant (Row and Column must be Equal and Must more than 1)
    6.Inverse (Row and Column must be Equal and Must more than 1)
6
Row and Column must be Equal and Must more than 1
Please select again.
***** Welcome to the Matrix Calculator *****
Choose the menu :
    1.Square Matrix for Det and Inverse
    2.Enter You row and column
    3.Quit
```

ในที่นี้เรากำหนดแถวเป็น 2 และคอลัมน์เป็น 3
เมื่อเลือกเมนู Inverse โปรแกรมจะขึ้นข้อความ Please select again.
และกลับไปเมนูหลัก เนื่องจากแถวและคอลัมน์ไม่เท่ากัน


```
C:\Users\Hp\Desktop\Project cs\Project.exe
Enter number b [0][1]: 2
Enter number b [0][2]: 1
Enter number b [1][0]: 9
Enter number b [1][1]: 7
Enter number b [1][2]: 8
Enter number b [2][0]: 6
Enter number b [2][1]: 4
Enter number b [2][2]: 5

Entered matrix 1st:
9.00  8.00  7.00
6.00  4.00  5.00
1.00  2.00  3.00
Entered matrix 2nd:
3.00  2.00  1.00
9.00  7.00  8.00
6.00  4.00  5.00

Do you want to
1.Addition
2.Subtraction
3.Multiplication
4.Division
5.Determinant (Row and Column must be Equal and Must more than 1)
6. Inverse (Row and Column must be Equal and Must more than 1)
6

Inverse of 1st or 2nd
2

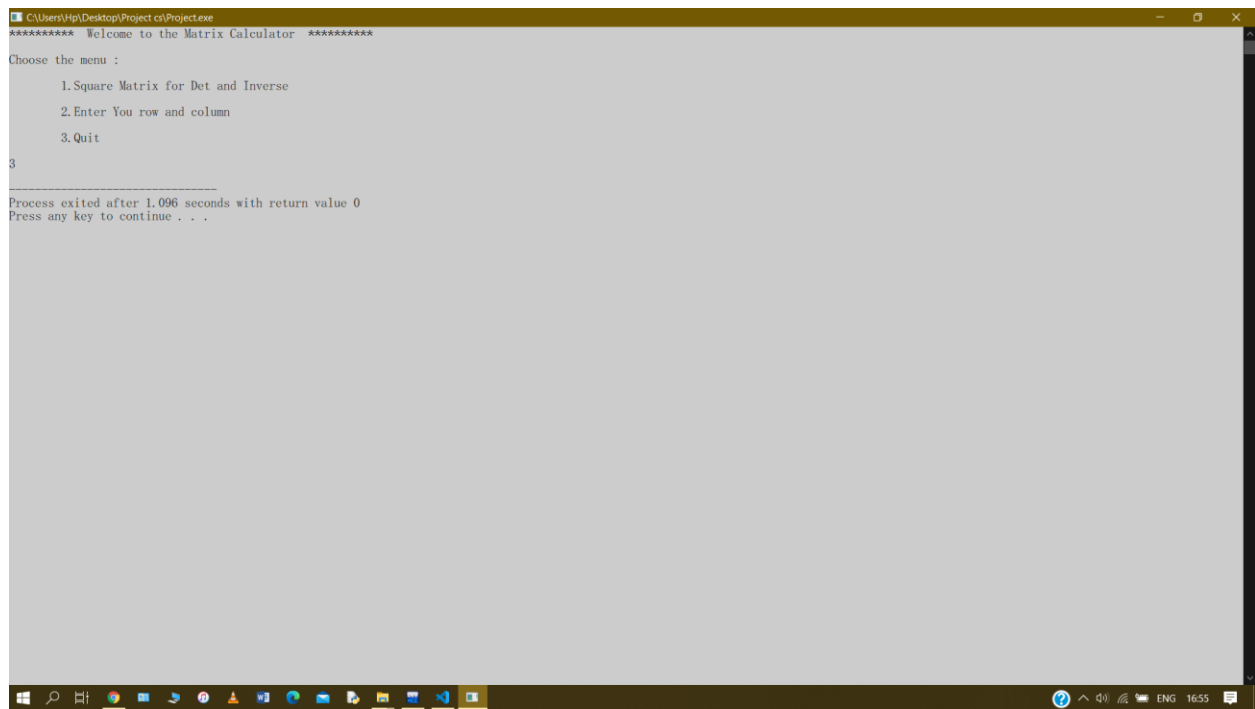
The inverse matrix is:
0.33  -0.67  1.00
0.33   1.00  -1.67
-0.67  0.00  0.33

Do you want to continue using the program? .(Y/N) :
```

เมื่อกำหนดแถวและหลักเท่ากันแล้ว หลังจากใส่ตัวเลขและเลือกเมนู Inverse โปรแกรมจะแสดงค่า Inverse ของเมทริกซ์ที่เราเลือก ในที่นี้เลือกหา Inverse ของเมทริกซ์ 2

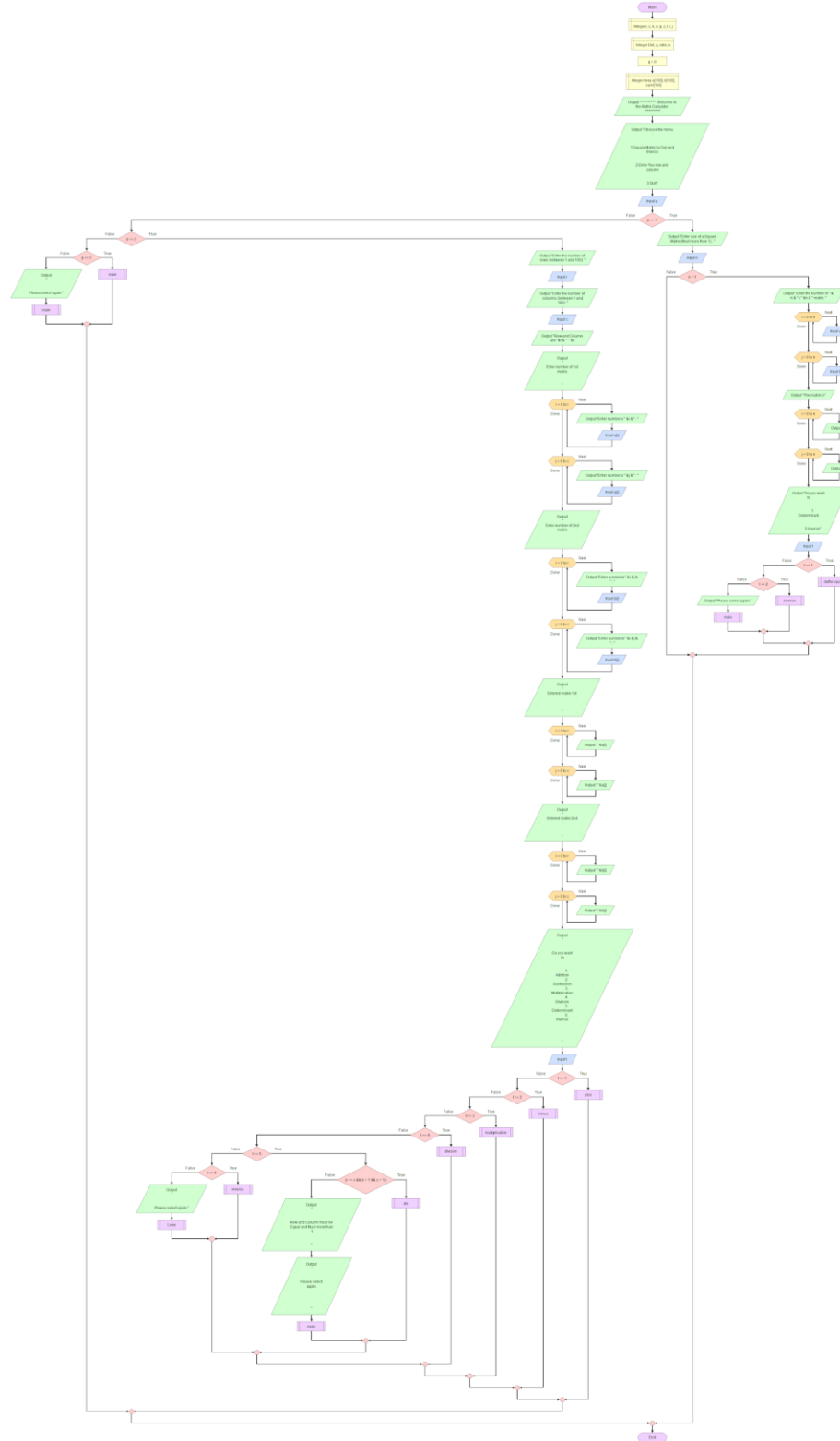
```
C:\Users\Hp\Desktop\Project cs\Project.exe
9.00  8.00  7.00
6.00  4.00  5.00
1.00  2.00  3.00
Entered matrix 2nd:
3.00  2.00  1.00
9.00  7.00  8.00
6.00  4.00  5.00
Do you want to
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Determinant (Row and Column must be Equal and Must more than 1)
6. Inverse (Row and Column must be Equal and Must more than 1)
6
Inverse of 1st or 2nd
2
The inverse matrix is:
0.33  -0.67  1.00
0.33  1.00  -1.67
-0.67  0.00  0.33
Do you want to continue using the program? .(Y/N) :
1
Do you want to continue using the program? .(Y/N) :
n
Thank You.
Process exited after 232.6 seconds with return value 0
Press any key to continue . . .
```

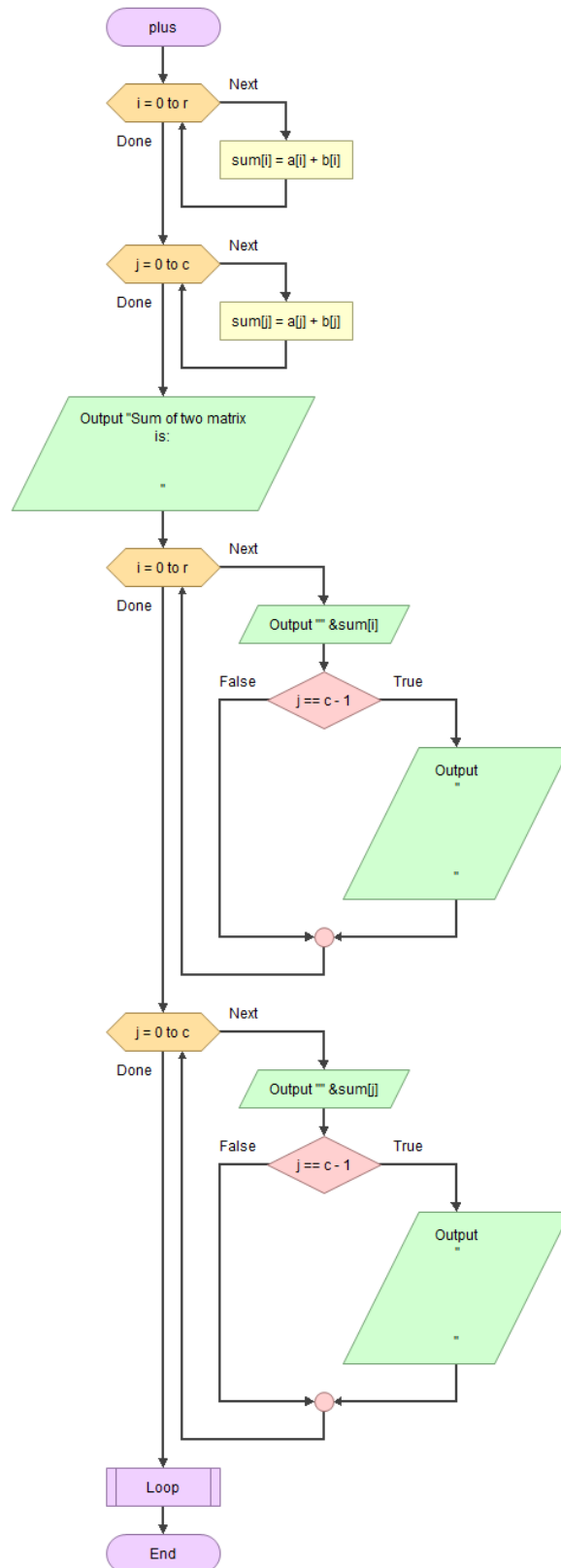
โปรแกรมจะถามว่าต้องการใช้งานโปรแกรมต่อหรือไม่
โดยถ้าต้องการใช้งานต่อให้กด y หรือ Y เพื่อกลับไปยังเมนูหลัก
ถ้าไม่ต้องการใช้งานต่อให้กด n หรือ N เพื่อออกจากโปรแกรม
หากใส่ นอกเหนือจากที่กล่าวมา
โปรแกรมจะถามอีกครั้งว่าต้องการใช้งานต่อหรือไม่



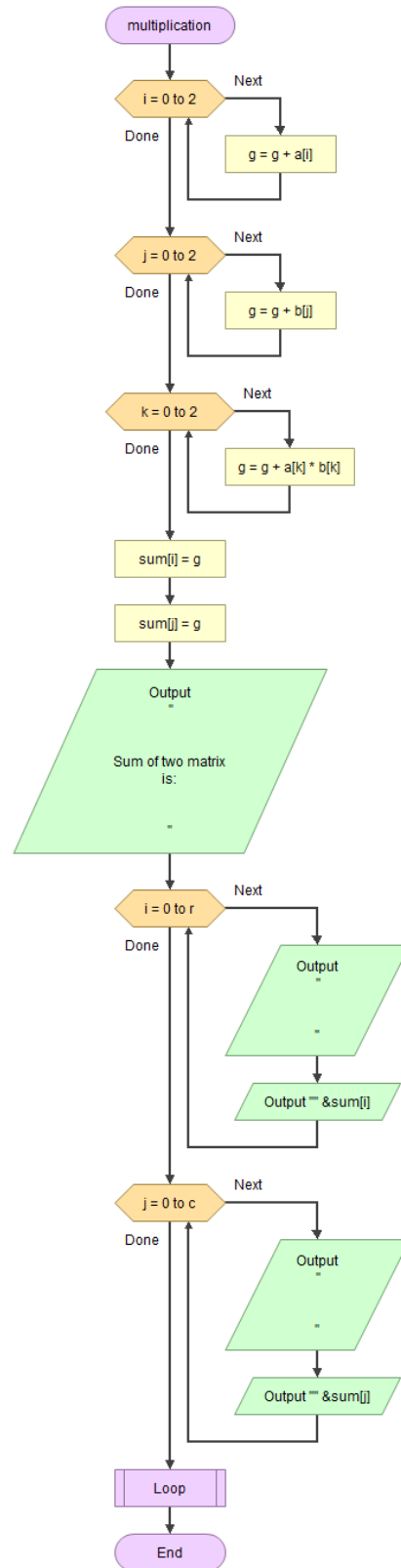
ที่เมนูหลัก เมื่อเลือกเมนูที่ 3 จะเป็นการออกจากโปรแกรม

Flowchart : Plus

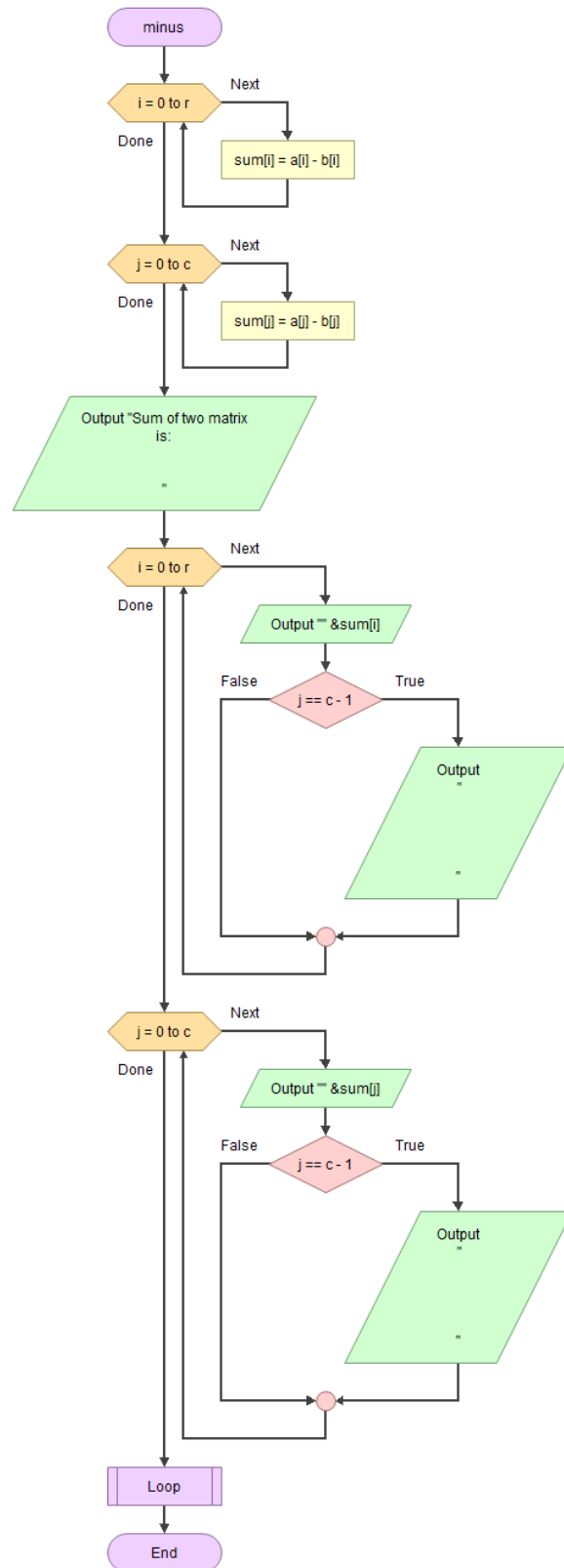




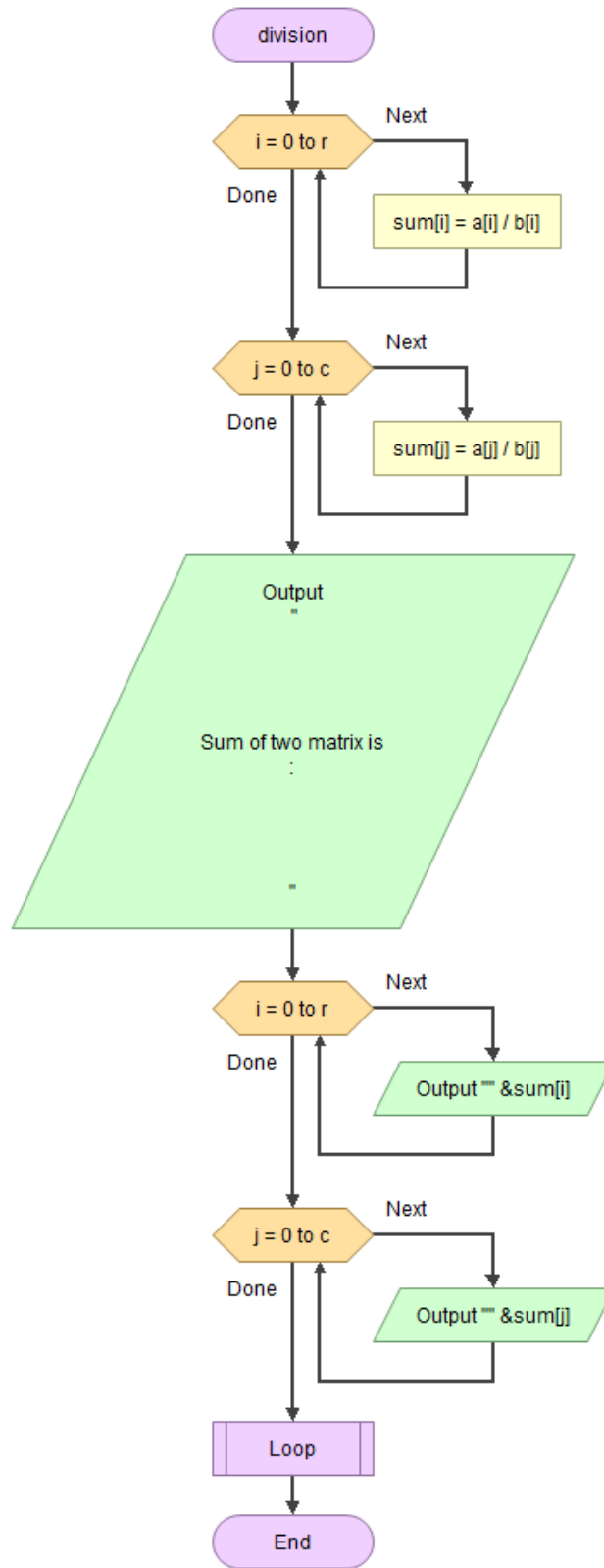
Flowchart : Multiplication



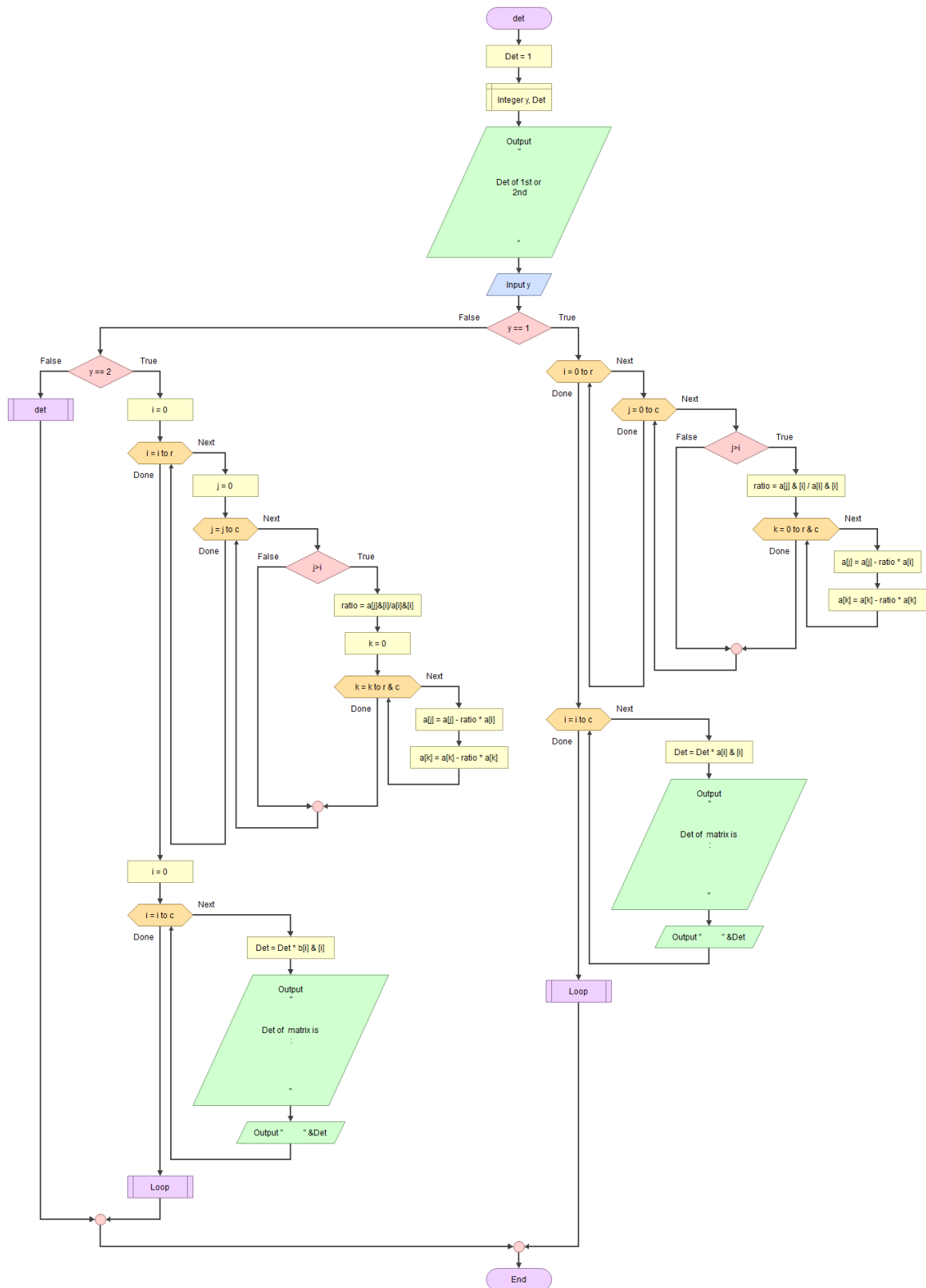
Flowchart : Minus



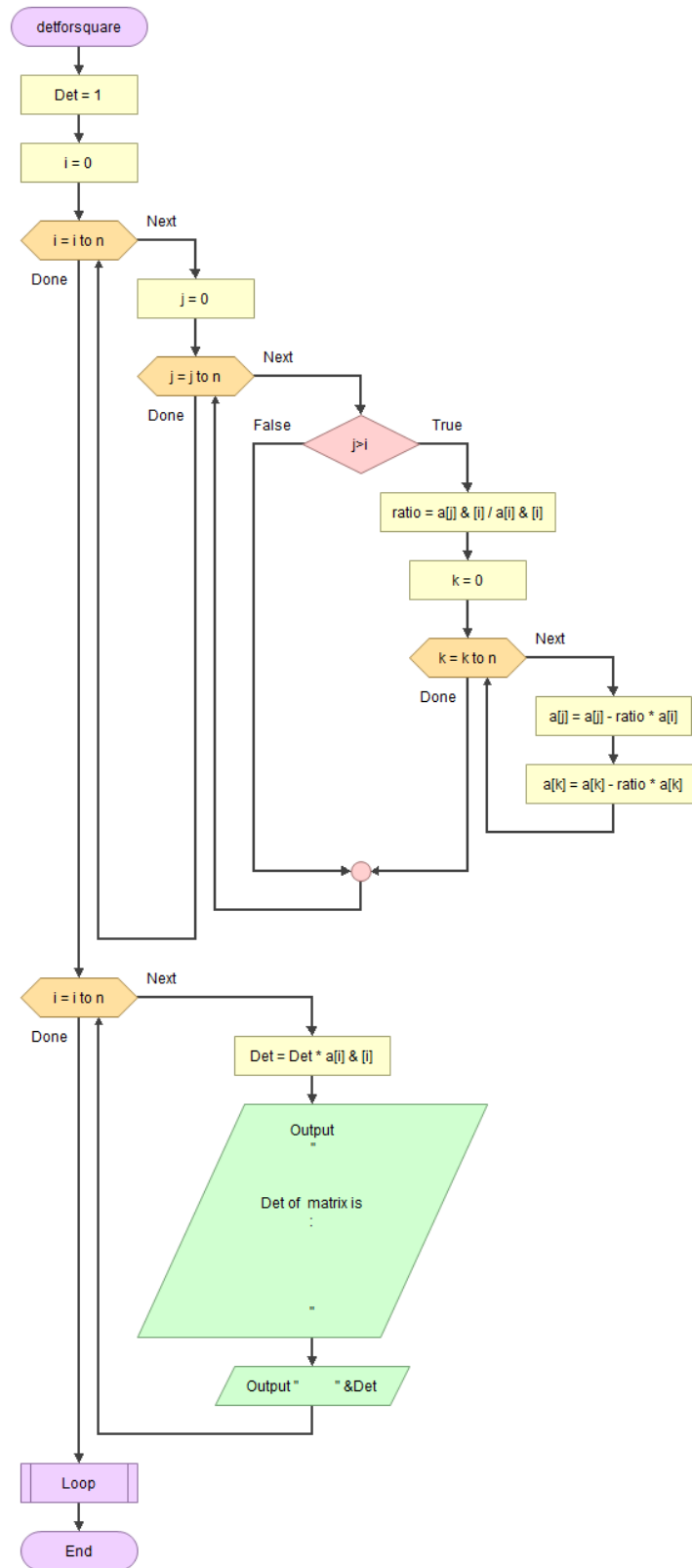
Flowchart : Division



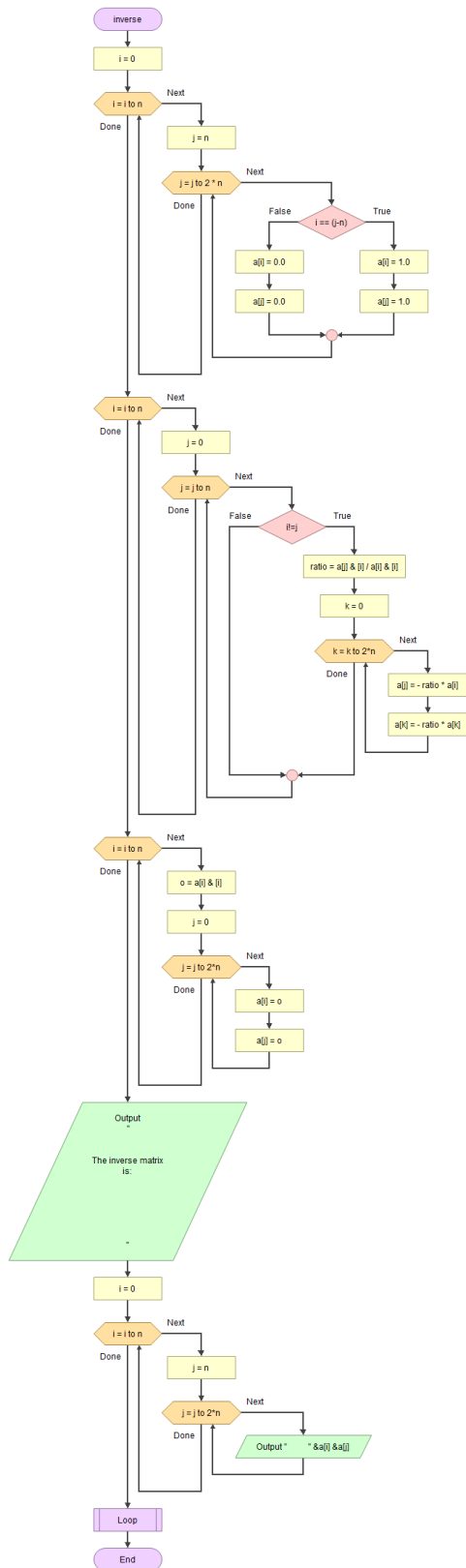
Flowchart : Det



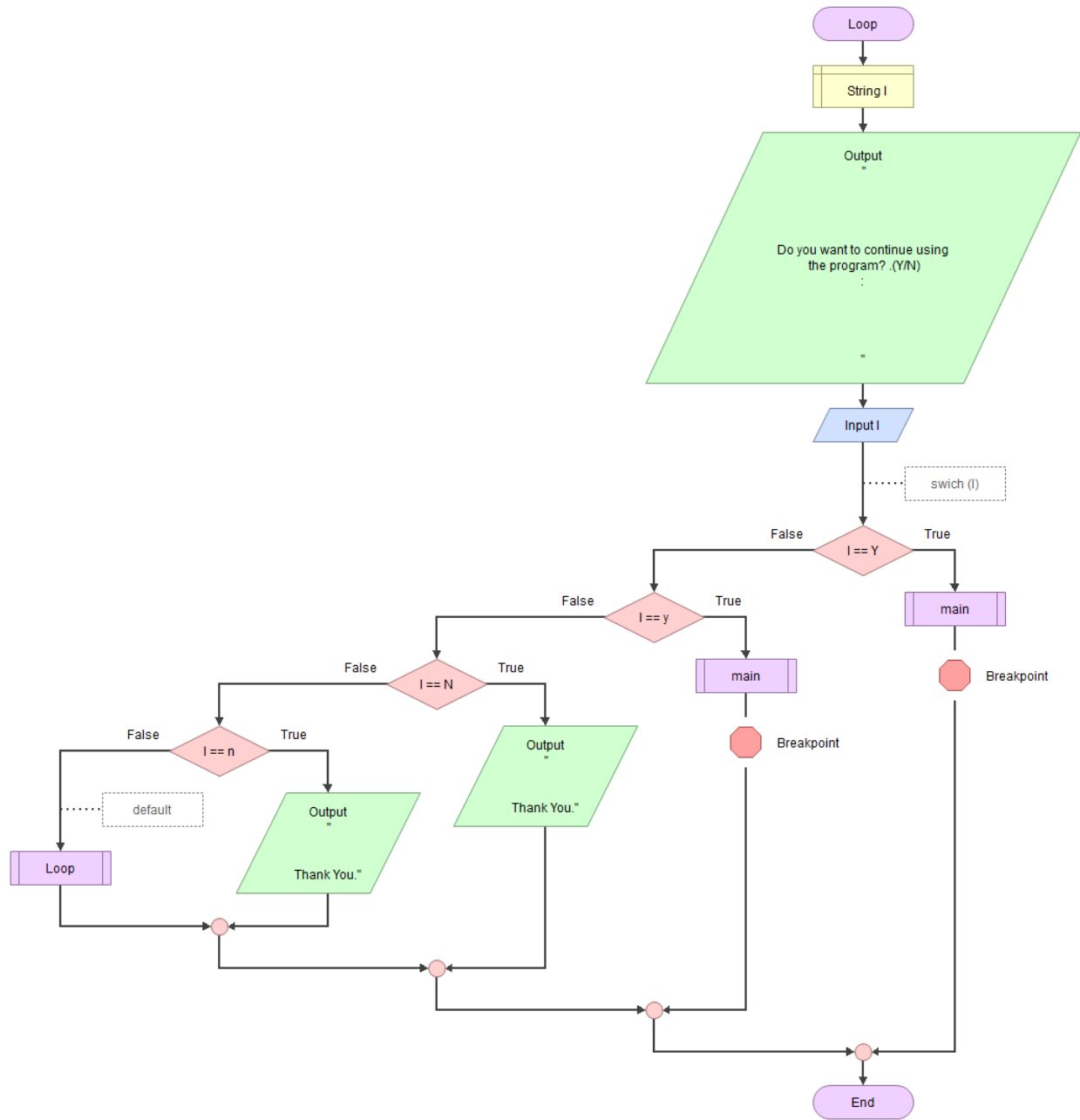
Flowchart : Det Foursquare



Flowchart : Inverse



Flowchart : Loop



Code

```

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int r, c, i, j, k, n, p, z;
float a[100][100], b[100][100], sum[100][100], Det, g= 0, ratio, o;
#####สร้างฟังก์ชันที่ต้องการใช้งาน#####
#####
void plus();
void multiplication();
void minus();
void division();
void det();
void detforsquare();
void inverse();
void inverseforsquare();
int Loop();
#####ส่วนของหน้าหลัก#####
int main() {
    system("COLOR 70");  /// เปลี่ยนสีพื้นหลังและตัวอักษร
    int t;
    printf("***** Welcome to the Matrix Calculator
*****\n\n");
    printf("Choose the menu :\n\n\t1.Square Matrix for Det and
Inverse\n\n\t2.Enter You row and column\n\n\t3.Quit\n\n");
    scanf("%d", &p); /// เก็บค่าจากตัวเลือก
    if (p == 1)
    {
        system("COLOR e0");
        printf("\nEnter size of a Square Matrix (Must more than 1) : \n\n");
        scanf("%d", &n);

        if (n > 1){
            printf("\nEnter the number of %d x %d matrix: \n" ,n,n);
            for(i = 0 ;i < n;i++)
                for(j = 0;j < n;j++)
                    scanf("%f", &a[i][j]);  ////เก็บค่าไว้ที่อาร์เรย์

            printf("\nThe matrix is\n");
            for(i = 0;i < n; i++){

```

```

printf("\n");
for(j = 0; j < n; j++)
    printf("%.2f\t", a[i][j]);
}

printf("\n\nDo you want
to\n\n\t1.Determinant\n\n\t2.Inverse\n\n");
scanf ("%d",&t);
if (t == 1)
{
    detforsquare(); /// เข้าสู่วิธีที่เรารว
}
else if (t == 2)
{
    inverseforsquare();
}
else {
    printf("\nPlease select again.\n\n");
    return main(); /// กลับไปที่หน้าหลัก
}

else {
    printf("\nPlease select again.\n\n");
    return main();
}

}

else if (p == 2){
system("COLOR e9");
printf("\nEnter the number of rows (between 1 and 100): ");
scanf("%d", &r);
printf("\nEnter the number of columns (between 1 and 100): ");
scanf("%d", &c);
printf("\nRow and Column are %d %d\n",r,c);
printf("\nEnter number of 1st matrix:\n");
for (i = 0; i < r; i++)
    for (j = 0; j < c; j++) {
        printf("Enter number a [%d][%d]: ", i , j );
        scanf("%f", &a[i][j]);
    }

printf("Enter number of 2nd matrix:\n");

```

```

for (i = 0; i < r; i++)
    for (j = 0; j < c; j++) {
        printf("Enter number b [%d][%d]: ", i , j);
        scanf("%f", &b[i][j]);
    }
printf("\nEntered matrix 1st: \n");
for (i = 0; i < r; i++){

    printf("\n");
    for (j = 0; j < c; j++) {
        printf("%.2f\t", a[i][j]);

    }
}
printf("\nEntered matrix 2nd: \n");
for (i = 0; i < r; i++){

    printf("\n");
    for (j = 0; j < c; j++) {
        printf("%.2f\t", b[i][j]);

    }
}
printf("\n\nDo you want to
\n\n\t1.Addition\n\n\t2.Subtraction\n\n\t3.Multiplication\n\n\t4.Division\
\n\n\t5.Determinant (Row and Column must be Equal and Must more than
1)\n\n\t6.Inverse (Row and Column must be Equal and Must more than
1)\n\n");
scanf ("%d",&t);
if (t == 1)
{
    plus();
}
else if (t == 2)
{
    minus();
}
else if (t == 3)
{
    multiplication();
}

```

```

    }
    else if (t == 4)
    {
        division();
    }
    else if (t == 5)
    {
        if (r == c && (r > 1 && c > 1)){
            det();
        }
        else{
            printf("\nRow and Column must be Equal and Must more than
1\n");

            printf("\nPlease select again.\n\n");
            return main();
        }

    }

    else if (t == 6)
    {
        if (r == c && (r > 1 && c > 1)){
            inverse();
        }
        else{
            printf("\nRow and Column must be Equal and Must more than
1\n");

            printf("\nPlease select again.\n\n");
            return main();
        }

    }

    else {

        printf("\nPlease select again.");
        Loop ();
    }

}

else if (p == 3){
    return 0;
}

```



```

    }
    else {
        printf("\nPlease select again.");
        return main();
    }
}

//////////ส่วนของฟังก์ชันบวก//////////
void plus()
{

    for (i = 0; i < r; i++)
        for (j = 0; j < c; j++) {
            sum[i][j] = a[i][j] + b[i][j];
        }

    printf("\nSum of two matrix is: \n");
    for (i = 0; i < r; i++){
        printf("\n");
        for (j = 0; j < c; j++) {
            printf("%.2f  ", sum[i][j]);
        }
    }

    Loop ();
}

//////////ส่วนของฟังก์ชันลบ//////////
void minus()
{

    for (i = 0; i < r; i++)
        for (j = 0; j < c; j++) {
            sum[i][j] = a[i][j] - b[i][j];
        }

    printf("\nSum of two matrix is: \n");
    for (i = 0; i < r; i++){

        printf("\n");
        for (j = 0; j < c; j++) {

```

```

        printf("%.2f\t", sum[i][j]);
    }

}

Loop (); //เมื่อแสดงค่าเสร็จให้ไปที่ฟังก์ชัน Loop

}

//////////ส่วนของฟังก์ชันคูณ//////////
void multiplication()
{
    for(i = 0; i < r; i++){

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

            sum[i][j] = 0;

            for(k = 0; k < r && k < c; k++){

                sum[i][j] = sum[i][j] + (a[i][k] * b[k][j]);

            }

        }

    }

    printf("\nSum of two matrix is: \n");
    for (i = 0; i < r; i++){

        printf("\n");
        for (j = 0; j < c; j++) {
            printf("%.2f\t", sum[i][j]);
        }

    }

    Loop ();
}

//////////ส่วนของฟังก์ชันหาร//////////
void division()
{
    for (i = 0; i < r; i++)
        for (j = 0; j < c; j++){

```

```

        sum[i][j] = a[i][j]/b[i][j];
    }

printf("\n\nSum of two matrix is : \n");
for (i = 0; i < r; i++){

    printf("\n");
    for (j = 0; j < c; j++) {
        printf("%.2f\t", sum[i][j]);
    }
}
Loop ();
}

//////////ส่วนของฟังก์ชัน Det สำหรับกำหนดแถวและคอลัมน์เอง//////////
void det()
{
    Det = 1;
    int y;
    printf("\nDet of 1st or 2nd \n\n");
    scanf ("%d", &y);
    if (y == 1){

        for(i = 0; i < r; i++){
            for(j = 0; j < c; j++){
                if(j>i){
                    ratio = a[j][i]/a[i][i];
                    for(k = 0; k < r && k < c; k++){
                        a[j][k] = a[j][k] - ratio * a[i][k];
                    }
                }
            }
        }

        for(i = 0; i < c; i++)
            Det = Det * a[i][i];
        printf("\nDet of matrix is :\n\n");
        printf("\t");
        printf("%.2f ", Det);

        Loop();
    }
}

```

```

}

else if (y == 2){

    for(i = 0; i < r; i++){
        for(j = 0; j < c; j++){
            if(j>i){
                ratio = a[j][i]/a[i][i];
                for(k = 0; k < r && k < c; k++){
                    a[j][k] = a[j][k] - ratio * a[i][k];
                }
            }
        }
    }

    for(i = 0; i < c; i++)
        Det = Det * b[i][i];
    printf("\nDet of matrix is :\n\n");
    printf("\t");
    printf("%.2f ", Det);

    Loop();
}

else{
    system("COLOR e4");
    printf("\nPlease select again.\n");
    return det();
}
}

//////////ส่วนของฟังก์ชัน Det สำหรับเมทริกซ์จัตุรัส//////////
void detforsquare()
{
    Det = 1;
    for(i = 0; i < n; i++){
        for(j = 0; j < n; j++){
            if(j>i){
                ratio = a[j][i]/a[i][i];
                for(k = 0; k < n; k++){
                    a[j][k] = a[j][k] - ratio * a[i][k];
                }
            }
        }
    }
}

```

```

    }
}

}

}

for(i = 0; i < n; i++)
    Det = Det * a[i][i];
printf("\nDet of matrix is :\n\n");
printf("\t");
printf("%.2f ", Det);

    Loop();
}

//////////ส่วนของฟังก์ชัน inverse สำหรับกำหนดแถวและคอลัมน์เอง//////////
void inverse()
{
    int u;
    printf("\nInverse of 1st or 2nd \n\n");
    scanf ("%d", &u);
    if (u == 1){
        for(i = 0; i < r; i++){
            for(j = c; j < 2*c; j++){
                if(i==(j-c))
                    a[i][j] = 1.0;
                else
                    a[i][j] = 0.0;
            }
        }
        for(i = 0; i < r; i++){
            for(j = 0; j < c; j++){
                if(i!=j){
                    ratio = a[j][i]/a[i][i];
                    for(k = 0; k < 2*r && k < 2*c; k++){
                        a[j][k] -= ratio * a[i][k];
                    }
                }
            }
        }
    }
    for(i = 0; i < r; i++){
        o = a[i][i];
    }
}

```

```

        for(j = 0; j < 2*c; j++){
            a[i][j] /= o;
        }
    }
    printf("\n\tThe inverse matrix is: \n\n");
    for(i = 0; i < r; i++){
        printf("\t");
        for(j = c; j < 2*c; j++){
            printf("%.2f", a[i][j]);
            printf("\t");
        }
        printf("\n");
    }
}

else if (u == 2){
    for(i = 0; i < r; i++){
        for(j = c; j < 2*c; j++){
            if(i==(j-c))
                b[i][j] = 1.0;
            else
                b[i][j] = 0.0;
        }
    }

    for(i = 0; i < r; i++){
        for(j = 0; j < c; j++){
            if(i!=j){
                ratio = b[j][i]/b[i][i];
                for(k = 0; k < 2*r && k < 2*c; k++){
                    b[j][k] -= ratio * b[i][k];
                }
            }
        }
    }

    for(i = 0; i < r; i++){
        o = b[i][i];
        for(j = 0; j < 2*c; j++){
            b[i][j] /= o;
        }
    }

    printf("\n\tThe inverse matrix is: \n\n");

```

```

for(i = 0; i < r; i++){
    printf("\t");
    for(j = c; j < 2*c; j++){
        printf("%.2f", b[i][j]);
        printf("\t");
    }
    printf("\n");
}
}

else {
    system("COLOR e4");
    printf("\nPlease select again,\n");
    return inverse();
}

Loop();
}

//////////ส่วนของฟังก์ชัน inverse สำหรับเมทริกซ์จัตุรัส//////////
void inverseforsquare()
{

    for(i = 0; i < n; i++){
        for(j = n; j < 2*n; j++){
            if(i==(j-n))
                a[i][j] = 1.0;
            else
                a[i][j] = 0.0;
        }
    }

    for(i = 0; i < n; i++){
        for(j = 0; j < n; j++){
            if(i!=j){
                ratio = a[j][i]/a[i][i];
                for(k = 0; k < 2*n; k++){
                    a[j][k] -= ratio * a[i][k];
                }
            }
        }
    }

    for(i = 0; i < n; i++){
        o = a[i][i];

```

```

        for(j = 0; j < 2*n; j++){
            a[i][j] /= o;
        }
    }
    printf("\n\tThe inverse matrix is: \n\n");
    for(i = 0; i < n; i++){
        printf("\t");
        for(j = n; j < 2*n; j++){
            printf("%.2f", a[i][j]);
            printf("\t");
        }
        printf("\n");
    }
    Loop();
}

//////////ส่วนของฟังก์ชัน Loop//////////
int Loop ()
{
    system("COLOR b0");
    char l;

    printf("\n\nDo you want to continue using the program? .(Y/N) :
\n\n");
    scanf("%s",&l);

    switch (l) {
        case 'Y':
            system("cls");  ///// ฟังก์ชันล้างหน้าจอ /////
            return main();
            break;
        case 'y':
            system("cls");
            return main();
            break;
        case 'N':
            system("COLOR c");
            printf("\nThank You.");
            return 0;  ///// ออกจากโปรแกรม /////
        case 'n':
            system("COLOR c");
            printf("\nThank You.");

```



```
        return 0;  
    default:  
        Loop();  
    }  
}
```