NAME TANISHQ GARG
ROLL NUMBER IIT2024038
PROJECT 2 , QUESTION 3

ALGORITHM :

Input:

1. A matrix of coefficients `coeff[n][n]` representing the system of equations.
2. A matrix of constants `cons[n][1]` representing the right-hand side of the equations.
3. The size of the system (number of variables) `n`.

Output:

4. The determinant of the coefficient matrix (Δ).
5. Values of the variables if a unique solution exists, otherwise a message stating "no solution."

Steps:

6. Input the number of variables `n`.
   - Prompt the user to enter the number of variables in the system.
7. Input the coefficient matrix `coeff[n][n]`.
   - Loop through each element of the coefficient matrix and read it from the user.
8. Input the constants matrix `cons[n][1]`.
   - Loop through each element of the constants matrix and read it from the user.
9. Display the coefficient matrix.
   - Print the inputted coefficient matrix for the user to see.
10. Calculate the determinant of the coefficient matrix (Δ).
    - Call the function `determinant(coeff, n)` to calculate the determinant of the coefficient matrix using recursive cofactor expansion.
11. Check if the determinant (Δ) is zero:
    - If Δ = 0, print "no solution" because the system of equations has no unique solution (it may have infinite solutions or be inconsistent).
    - If Δ ≠ 0, proceed to calculate the solutions using Cramer's rule.

12. For each variable `p` (from 1 to n):
    - Generate the modified matrix for the `p`-th variable:
        - For each row `i`:
            - Replace the `p`-th column of the coefficient matrix with the constants from the `cons` matrix.
        - Display the modified matrix for the `p`-th variable.
    - Calculate the determinant of the modified matrix:
        - Call the function `determinant()` on the modified matrix for the `p`-th variable.
    - Compute the value of the variable:
        - Calculate the value of the `p`-th variable as the ratio of the determinant of the modified matrix to the determinant of the original coefficient matrix:

          Variable p = determinant of modified matrix p/ determinant of delta matrix
          Display the value of the `p`-th variable.

Functions:

13. getCofactor(matrix, temp, p, q, n):
    - Compute the cofactor of the matrix by excluding row `p` and column `q`.
    - Store the resulting minor matrix in `temp`.
14. determinant(matrix, n):
    - Base case: If the matrix is 1x1, return the single element.
    - Recursive case: For a matrix of size `n`, expand along the first row, calculate the cofactor of each element, and recursively compute the determinant of the smaller matrix.

1. Multiply each cofactor's determinant by the corresponding matrix element and alternating sign, and sum these values to get the determinant.

CODE :

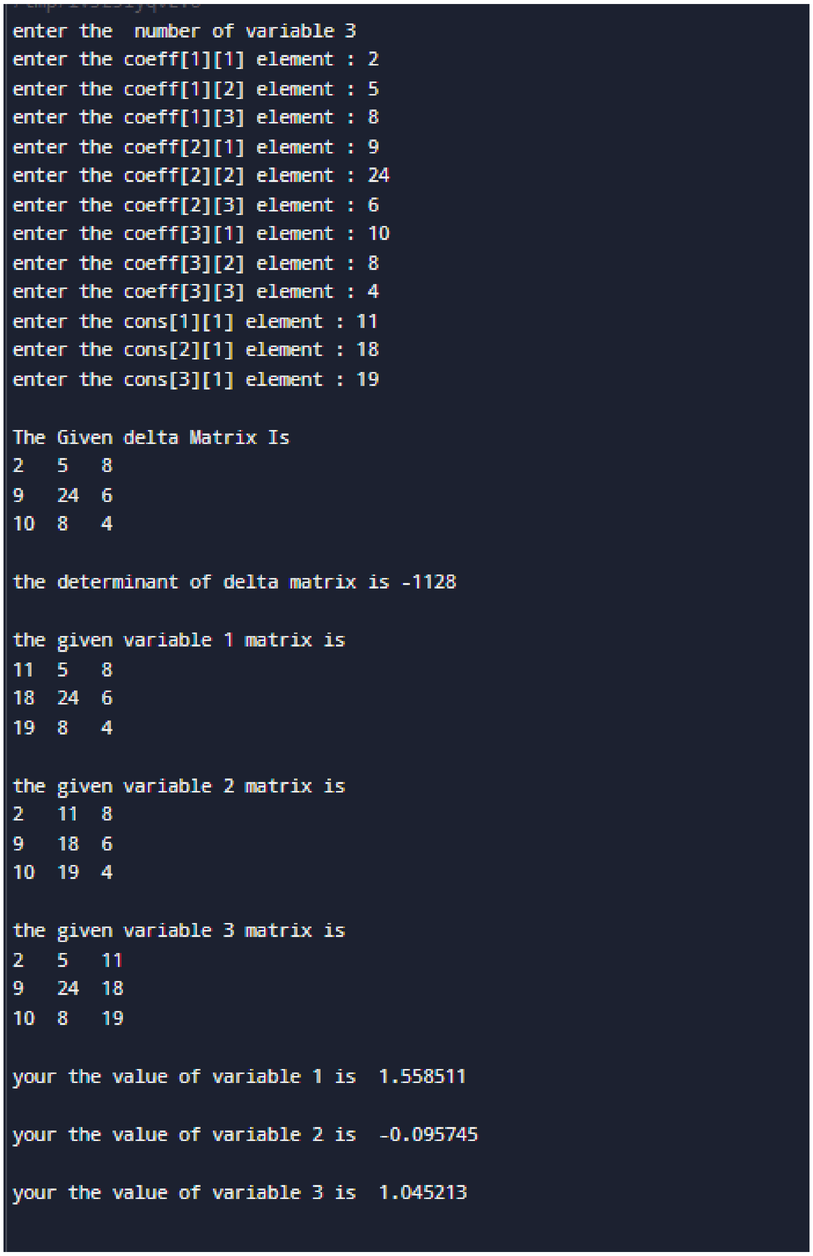
```
#include <stdio.h>
#include <string.h>
#define MAX 10
void getCofactor(int matrix[MAX][MAX], int temp[MAX][MAX], int p, int q, int n)
{
   int i = 0, j = 0;
   for (int row = 0; row < n; row++)
```

```
    {
        for (int col = 0; col < n; col++)
        {
            if (row != p && col != q)
            {
                temp[i][j++] = matrix[row][col];
                if (j == n - 1)
                {
                    j = 0;
                    i++;
                }
            }
        }
    }
}
int determinant(int matrix[MAX][MAX], int n)
{
    if (n == 1)
    {
        return matrix[0][0];
    }
    int temp[MAX][MAX];
    int det = 0;
    int sign = 1;
    for (int f = 0; f < n; f++)
    {
        getCofactor(matrix, temp, 0, f, n);
        det += sign * matrix[0][f] * determinant(temp, n - 1);
        sign = -sign;
    }
    return det;
}
int main()
{
    int n;
    printf("enter the  number of variable ");
    scanf("%d", &n);
    int coeff[MAX][MAX];
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
```

```
        printf("enter the coeff[%d][%d] element : ", i + 1, j + 1);
        scanf("%d", &coeff[i][j]);
    }
}
int cons[MAX][MAX];
for (int j = 0; j < n; j++)
{
    printf("enter the cons[%d][1] element : ", j + 1);
    scanf("%d", &cons[j][1]);
}
printf("\nThe Given delta Matrix Is\n");
for (int i = 0; i < n; i++)
{
    for (int j = 0; j < n; j++)
    {
        printf("%d\t", coeff[i][j]);
    }
    printf("\n");
}
printf("\nthe determinant of delta matrix is %d\n", determinant(coeff, n));
int a;
a = determinant(coeff, n);
if (a == 0)
{
    printf("there is no solution");
}
else
{
    for (int p = 0; p < n; p++)
    {
        printf("\nthe given variable %d matrix is\n", p + 1);
        for (int i = 0; i < n; i++)
        {
            for (int j = 0; j < n; j++)
            {
                if (j == p)
                {
                    printf("%d\t", cons[i][1]);
                }
                else
                {
                    printf("%d\t", coeff[i][j]);
                    }
                }
                printf("\n");
            }
        }
        for (int p = 0; p < n; p++)
        {
            int tempmat[MAX][MAX];
            memcpy(tempmat, coeff, sizeof(coeff));
            for (int i = 0; i < n; i++)
            {
                for (int j = 0; j < n; j++)
                {
                    if (j == p)
                    {
                        tempmat[i][j] = cons[i][1];
                    }
                }
            }
            printf("\nyour the value of variable %d is  %f\n", p+1,(float)determinant(tempmat, n) / a);

            return 0;
}
```

RESULT :



```
enter the  number of variable 3
enter the coeff[1][1] element : 2
enter the coeff[1][2] element : 5
enter the coeff[1][3] element : 8
enter the coeff[2][1] element : 9
enter the coeff[2][2] element : 24
enter the coeff[2][3] element : 6
enter the coeff[3][1] element : 10
enter the coeff[3][2] element : 8
enter the coeff[3][3] element : 4
enter the cons[1][1] element : 11
enter the cons[2][1] element : 18
enter the cons[3][1] element : 19

The Given delta Matrix Is
2   5   8
9   24  6
10  8   4

the determinant of delta matrix is -1128

the given variable 1 matrix is
11  5   8
18  24  6
19  8   4

the given variable 2 matrix is
2   11  8
9   18  6
10  19  4

the given variable 3 matrix is
2   5   11
9   24  18
10  8   19

your the value of variable 1 is  1.558511

your the value of variable 2 is  -0.095745

your the value of variable 3 is  1.045213
```