THIS RECORD IS DONE BY:

LAHARI(F6)
RASAGNA(F7)
AKSHAYA(E9)
SRUTHI(H3)

OF CLASS OF CSE-C FOR PPS LAB

DATE: **SIGNATURE:**

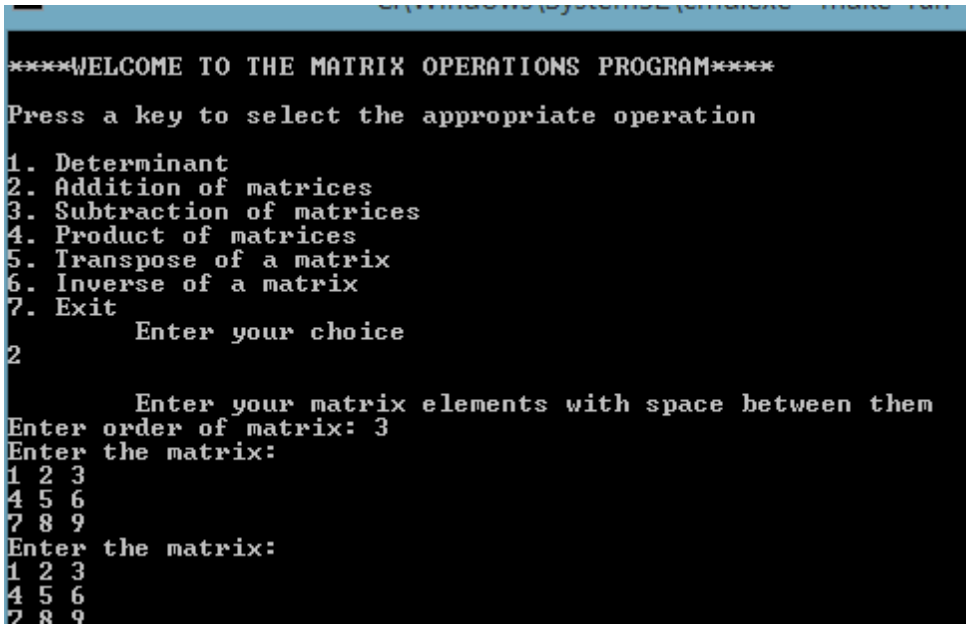
PROJECT STATEMENT:

MATRIX CALCULATOR

Mathematical operations are a part of our daily lives. Everyday we will be involved with various types of calculations around us. Matrices is a mathematical entity where numbers are arranged in rows and columns. A lot of applications exist with matrices in real life. The major application lies in the software industry such as development of algorithms like path finder algorithms, image processing algorithms and many more.

- In this project, some of the basic matrix operations are presented where a user can select the operation to be performed on the matrix.
- Then the matrices with their size are entered. Note that only square matrices are being considered for the project.
- Following Operations to be performed:
    1. Options to select matrix operation.
    2. Operations on two matrices such as addition, subtraction and multiplication include single matrix operations such as determinant, transpose and inverse of a matrix.
    3. A separate function for each operation structure has been implemented for storing the matrices.
    4. There is no upper limit for the size of the matrix.

TASK:



SOURCE CODE:

```
#include<stdio.h>

void read(int r, int c, int x[10][10]);
int dis(int r, int c, int x[10][10]);
void add(int r, int c, int x[10][10], int y[10][10], int result[10][10]);
void sub(int r, int c, int x[10][10], int y[10][10], int result[10][10]);
void transpose(int r, int c, int x[10][10], int result[10][10]);
void multiply(int x[10][10], int y[10][10], int r1, int c1, int c2, int result[10][10]);

int main() {
    int choice;
    int a[10][10], b[10][10], d[10][10];
    int r1, c1, r2, c2;

    printf("**WELCOME TO MATRIX OPERATIONS PROGRAM**\n");
    printf("Enter your option\n");
    printf("1. Addition of matrices\n");
    printf("2. Subtraction of matrices\n");
    printf("3. Multiplication of matrices\n");
    printf("4. Transpose of matrix\n");
    printf("Enter choice: ");
    scanf("%d", &choice);

    if (choice == 1 || choice == 2 || choice == 3) {
        printf("Enter the number of rows and columns of the 1st matrix:\n");
        scanf("%d,%d", &r1, &c1);
        printf("Enter elements of the 1st matrix:\n");
        read(r1, c1, a);

        printf("Enter the number of rows and columns of the 2nd matrix:\n");
        scanf("%d,%d", &r2, &c2);
        printf("Enter elements of the 2nd matrix:\n");
        read(r2, c2, b);
    } else if (choice == 4) {
        printf("Enter the number of rows and columns of the matrix:\n");
```

```
        scanf("%d,%d", &r1, &c1);
        printf("Enter elements of the matrix:\n");
        read(r1, c1, a);
    } else {
        printf("Invalid choice\n");
        return 1;  // Return an error code
    }

    switch(choice) {
        case 1:
            if (r1 == r2 && c1 == c2) {
                printf("Addition:\n");
                add(r1, c1, a, b, d);
                dis(r1, c1, d);
            } else {
                printf("Error: Matrices must have the same dimensions\n");
            }
            break;
        case 2:
            if (r1 == r2 && c1 == c2) {
                printf("Subtraction:\n");
                sub(r1, c1, a, b, d);
                dis(r1, c1, d);
            } else {
                printf("Error: Matrices must have the same dimensions\n");
            }
            break;
        case 3:
            if (c1 == r2) {
                multiply(a, b, r1, c1, c2, d);
                printf("Multiplication of given two matrices is:\n");
                dis(r1, c2, d);
            } else {
                printf("Error: Number of columns in the 1st matrix must be equal to the
number of rows in the 2nd matrix\n");
            }
            break;
        case 4:
            transpose(r1, c1, a, d);
            dis(c1, r1, d);
```

```
            break;
        default:
            printf("Invalid choice\n");
            break;
    }

    return 0;
}

void read(int r, int c, int x[10][10]) {
    int i, j;
    for(i = 0; i < r; i++) {
        for(j = 0; j < c; j++) {
            scanf("%d", &x[i][j]);
        }
    }
}

int dis(int r, int c, int x[10][10]) {
    int i, j;
    for(i = 0; i < r; i++) {
        for(j = 0; j < c; j++) {
            printf("%d ", x[i][j]);
        }
        printf("\n");
    }
    return 0;  // Return 0 for successful execution
}

void add(int r, int c, int x[10][10], int y[10][10], int result[10][10]) {
    int i, j;
    for(i = 0; i < r; i++) {
        for(j = 0; j < c; j++) {
            result[i][j] = x[i][j] + y[i][j];
        }
    }
}

void sub(int r, int c, int x[10][10], int y[10][10], int result[10][10]) {
    int i, j;
```

```
    for(i = 0; i < r; i++) {
        for(j = 0; j < c; j++) {
            result[i][j] = x[i][j] - y[i][j];
        }
    }
}

void transpose(int r, int c, int x[10][10], int result[10][10]) {
    int i, j;
    for(i = 0; i < r; i++) {
        for(j = 0; j < c; j++) {
            result[i][j] = x[j][i];
        }
    }
}

void multiply(int x[10][10], int y[10][10], int r1, int c1, int c2, int result[10][10]) {
    int i, j, k;
    for(i = 0; i < r1; i++) {
        for(j = 0; j < c2; j++) {
            result[i][j] = 0;
            for(k = 0; k < c1; k++) {
                result[i][j] += x[i][k] * y[k][j];
            }
        }
    }
}
```

OUTPUT:

```
**WELCOME TO MATRIX OPERATIONS PROGRAM**
Enter your option
1. Addition of matrices
2. Subtraction of matrices
3. Multiplication of matrices
4. Transpose of matrix
Enter choice: 1
Enter the number of rows and columns of the 1st matrix:
2,2
Enter elements of the 1st matrix:
1
2
3
4
Enter the number of rows and columns of the 2nd matrix:
2,2
Enter elements of the 2nd matrix:
1
2
3
4
Addition:
2 4
```

```
**WELCOME TO MATRIX OPERATIONS PROGRAM**
Enter your option
1. Addition of matrices
2. Subtraction of matrices
3. Multiplication of matrices
4. Transpose of matrix
Enter choice: 2
Enter the number of rows and columns of the 1st matrix:
2,3
Enter elements of the 1st matrix:
1
4
7
8
3
2
Enter the number of rows and columns of the 2nd matrix:
2,3
Enter elements of the 2nd matrix:
1
8
5
2
9
5
Subtraction:
0 -4 2
6 -6 -3
```

```
**WELCOME TO MATRIX OPERATIONS PROGRAM**
Enter your option
1. Addition of matrices
2. Subtraction of matrices
3. Multiplication of matrices
4. Transpose of matrix
Enter choice: 4
Enter the number of rows and columns of the matrix:
2,2
Enter elements of the matrix:
2
3
6
5
2 6
3 5
```

```
**WELCOME TO MATRIX OPERATIONS PROGRAM**
Enter your option
1. Addition of matrices
2. Subtraction of matrices
3. Multiplication of matrices
4. Transpose of matrix
Enter choice: 3
Enter the number of rows and columns of the 1st matrix:
2,2
Enter elements of the 1st matrix:
1
4
5
6
Enter the number of rows and columns of the 2nd matrix:
2,2
Enter elements of the 2nd matrix:
8
4
5
2
Multiplication of given two matrices is:
28 12
70 32
```