#include <stdio.h>

#include <stdlib.h>

int\*\* createMatrix(int rows, int cols) {

int \*\*matrix = (int\*\*)malloc(rows \* sizeof(int\*));

for (int i = 0; i < rows; i++) {

matrix[i] = (int\*)malloc(cols \* sizeof(int));

}

return matrix;

}

// Function to input matrix

void inputMatrix(int \*\*matrix, int rows, int cols) {

printf("Enter elements (%dx%d):\n", rows, cols);

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

scanf("%d", &matrix[i][j]);

}

}

}

int\*\* multiplyMatrices(int \*\*A, int \*\*B, int r1, int c1, int c2) {

int \*\*C = createMatrix(r1, c2);

for (int i = 0; i < r1; i++) {

for (int j = 0; j < c2; j++) {

C[i][j] = 0;

for (int k = 0; k < c1; k++) {

C[i][j] += A[i][k] \* B[k][j];

}

}

}

return C;

}

void displayMatrix(int \*\*matrix, int rows, int cols) {

printf("Matrix (%dx%d):\n", rows, cols);

for (int i = 0; i < rows; i++) {

for (int j = 0; j < cols; j++) {

printf("%d ", matrix[i][j]);

}

printf("\n");

}

}

int main() {

int r1, c1, r2, c2;

printf("Enter rows and columns for Matrix A: ");

scanf("%d %d", &r1, &c1);

printf("Enter rows and columns for Matrix B: ");

scanf("%d %d", &r2, &c2);

if (c1 != r2) {

printf("Matrix multiplication not possible. Columns of A must equal rows of B.\n");

return 1;

}

int \*\*A = createMatrix(r1, c1);

int \*\*B = createMatrix(r2, c2);

printf("\nMatrix A:\n");

inputMatrix(A, r1, c1);

printf("\nMatrix B:\n");

inputMatrix(B, r2, c2);

int \*\*C = multiplyMatrices(A, B, r1, c1, c2);

printf("\nResult Matrix:\n");

displayMatrix(C, r1, c2);

// Free allocated memory

for (int i = 0; i < r1; i++) free(A[i]);

for (int i = 0; i < r2; i++) free(B[i]);

for (int i = 0; i < r1; i++) free(C[i]);

free(A);

free(B);

free(C);

return 0;

}