**8.MULTIPLY TWO MATRIX**

**RECURSIVE:-**

#include <stdio.h>

void multiply(int, int, int [][10], int, int, int [][10], int [][10]);

void display(int, int, int[][10])

int main()

{

int a[10][10], b[10][10], c[10][10] = {0};

int m1, n1, m2, n2, i, j, k;

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

scanf("%d%d", &m1, &n1);

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

scanf("%d%d", &m2, &n2);

if (n1 != m2)

{

printf("Matrix multiplication not possible.\n");

}

else

{

printf("Enter elements in Matrix A:\n");

for (i = 0; i < m1; i++)

for (j = 0; j < n1; j++)

{

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

}

printf("\nEnter elements in Matrix B:\n");

for (i = 0; i < m2; i++)

for (j = 0; j < n2; j++)

{

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

}

multiply(m1, n1, a, m2, n2, b, c);

}

printf("On matrix multiplication of A and B the result is:\n");

display(m1, n2, c);

}

void multiply (int m1, int n1, int a[10][10], int m2, int n2, int b[10][10], int c[10][10])

{

static int i = 0, j = 0, k = 0;

if (i >= m1)

{

return;

}

else if (i < m1)

{

if (j < n2)

{

if (k < n1)

{

c[i][j] += a[i][k] \* b[k][j];

k++;

multiply(m1, n1, a, m2, n2, b, c);

}

k = 0;

j++;

multiply(m1, n1, a, m2, n2, b, c);

}

j = 0;

i++;

multiply(m1, n1, a, m2, n2, b, c);

}

}

void display(int m1, int n2, int c[10][10])

{

int i, j;

for (i = 0; i < m1; i++)

{

for (j = 0; j < n2; j++)

{

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

}

printf("\n");

}

}

