NAME:	Eshan Bhuse
UID:	2021300013
SUBJECT	Design and analysis of algorithm
EXPERIMENT NO:	3
AIM:	**Experiment on Strassen's matrix multiplication**
ALGORITHM	1. Take inputs as matrix1 and matrix2 2. elements of matrix 1: a00, a01, a10, a11 elements of matrix 2: b00, b01, b10, b11 3. perform following addition: m1 = (a11 + a00) * (b11 + b00) m2 = (a10 + a11) * b00 m3 = (b01 – b11) * a00 m4 = (b10 – b00) * a11 m5 = (a00 + a01) * b11 m6 = (a10 – a00) * (b00 + b01) m7 = (a01 – a11) * (b10 + b11) C00= m1 + m4 – m5 + m7 C01 = m3 + m5 C10 = m2 + m4 C11 = m1 + m3 – m2 – m6 4. print the resultant matrix

THEORY:	# Strassen's Matrix Multiplication **Strassen** in 1969 gave an overview on how we can find the multiplication of two **2*2dimension matrices by the brute-force algorithm**. But by using the divide and conquer technique the overall complexity for the multiplication of two matrices has been reduced. This happens by decreasing the total number of multiplications performed at the expense of a slight increase in the number of additions. **Strassen** has used some formulas for multiplying the two 2*2dimension matrices where the number of multiplications is seven, additions and subtractions are is eighteen, and in brute force algorithm, there is eight number of multiplications and four addition. When the order **n** of matrix reaches infinity, the utility of Strassen's formula is shown by its asymptotic superiority. For example, let us consider two matrices **A** and **B** of **n*n** dimension, where **n** is a power of two. It can be observed that we can have four submatrices of order **n/2 * n/2** from **A**, **B,** and their product **C** where **C** is the resultant matrix of **A** and **B**.

	The procedure of Strassen's matrix multiplication Here is the procedure: 1. Divide a matrix of the order of 2*2 recursively until we get the matrix of order 2*2. 2. To carry out the multiplication of the 2*2 matrix, use the previous set of formulas. 3. Subtraction is also performed within these eight multiplications and four additions. 4. To find the final product or final matrix combine the result of two matrixes. **Formulas for Strassen's matrix multiplication.** Following are the formulae that are to be used for matrix multiplication. m1 = (a11 + a00) * (b11 + b00) m2 = (a10 + a11) * b00 m3 = (b01 – b11) * a00 m4 = (b10 – b00) * a11 m5 = (a00 + a01) * b11 m6 = (a10 – a00) * (b00 + b01) m7 = (a01 – a11) * (b10 + b11) C00= m1 + m4 – m5 + m7 C01 = m3 + m5 C10 = m2 + m4 C11 = m1 + m3 – m2 – m6 Here, C00, C01, C10, and C11 are the elements of the 2*2 matrix.

PROGRAM:

```
#include <stdio.h>
#include<stdlib.h>
#include<time.h>
int main()
{
int a[2][2],b[2][2],c[2][2],i,j;
int m1,m2,m3,m4,m5,m6,m7;
clock_t start,end;
double exe_time;
start=clock();
printf("Enter the 4 elements of first matrix: ");
for(i=0;i<2;i++)
for(j=0;j<2;j++)
scanf("%d",&a[i][j]);
printf("Enter the 4 elements of second matrix: ");
for(i=0;i<2;i++)
for(j=0;j<2;j++)
scanf("%d",&b[i][j]);

printf("\nThe first matrix is\n");
for(i=0;i<2;i++)
{
printf("\n");
for(j=0;j<2;j++)
printf("%d\t",a[i][j]);
}

printf("\nThe second matrix is\n");
for(i=0;i<2;i++){
printf("\n");
for(j=0;j<2;j++)
printf("%d\t",b[i][j]);
}
```

```c
printf("\nThe second matrix is\n");
for(i=0;i<2;i++){
printf("\n");
for(j=0;j<2;j++)
printf("%d\t",b[i][j]);
}
m1= (a[0][0] + a[1][1])*(b[0][0]+b[1][1]);
m2= (a[1][0]+a[1][1])*b[0][0];
m3= a[0][0]*(b[0][1]-b[1][1]);
m4= a[1][1]*(b[1][0]-b[0][0]);
m5= (a[0][0]+a[0][1])*b[1][1];
m6= (a[1][0]-a[0][0])*(b[0][0]+b[0][1]);
m7= (a[0][1]-a[1][1])*(b[1][0]+b[1][1]);

c[0][0]=m1+m4-m5+m7;
c[0][1]=m3+m5;
c[1][0]=m2+m4;
c[1][1]=m1-m2+m3+m6;
printf("\n After performing multiplication \n");
for(i=0;i<2;i++){
printf("\n");
for(j=0;j<2;j++)
printf("%d\t",c[i][j]);
}
end=clock();
exe_time=((double)(end-start))/CLOCKS_PER_SEC;
printf("\nThe time taken:%f",exe_time);
return 0;
}
```

```
Enter the 4 elements of first matrix: 1
2
3
4
Enter the 4 elements of second matrix: 1
2
3
4

The first matrix is

1       2
3       4
The second matrix is

1       2
3       4
 After performing multiplication

7       10
15      22
The time taken:0.000233

...Program finished with exit code 0
Press ENTER to exit console.
```

OBSERVATION:	Strassen’s matrix multiplication reduces the number of multiplication as it takes more time complexity than the addition and subtraction. So, operations are replaced by addition. For the big matrices, this method is very useful and fast. Time complexity of Strassen’s Matrix multiplication: **T(n) = 7T(n/2) + O(n^2) = O (n ^log (7)).**
CONCLUSION:	Successfully performed the experiment of Strassen’s matrix multiplication in C Language and observed that Strassen’s matrix multiplication is useful for higher order matrix multiplication and its time complexity is O (n^ log (7)).