 NAME-ANIL KUMAR

EN. NO-02711804421

DATE:-

EMAIL:-anilstmnet@gmail.com

INDEXING

|  |  |
| --- | --- |
| **S.N** | **QUESTION** |
| 01 | Multiplying the two square matrix. |
| 02 | Linear search |
| 03 | Bubble sort |
| 04 | Transpose a square matrix of n\*n without any using auxilary matrix. |
| 05 | Stack and its implementation using array. |
| 06 | Implementationg queue and its operation using array. |
| 07 | Factorial of N:-  i)using recurssion ii)iteration. |
| 08 | Fibonacci of N:-  i)recurssion ii)iteration. |
| 09 | Binary search. |
| 10 | Insertion sort. |
| 11 | Selection sort. |
| 12 | Linked list. |
| 13 | Merge sort. |
| 14 | Quick sort. |
| 15 | Implementating a queue using two stacks. |
| 16 | Preorder ,postorder and inorder triversal of binary search tree. |

1. Multiplying the two square matrix.

#include<stdio.h>

#include<conio.h>

int main()

{

    int sum,mat1[3][3],mat2[3][3],mult[3][3],i,j,k;

    printf("\n enter 9 value for mat1: "); //store the value in first array

    for(i=0;i<=2;i++)

    {

        for(j=0;j<=2;j++)

        {

            scanf("%d",&mat1[i][j]);

        }

    }

    printf("\n enter 9 value for mat2: ");//store the value in 2nd array

    for(i=0;i<=2;i++)

    {

        for(j=0;j<=2;j++)

        {

            scanf("%d",&mat2[i][j]);

        }

    }

    for(i=0;i<=2;i++)// multiplication logic

    {

        for(j=0;j<=2;j++)

        {

            sum=0;

            for(k=0;k<=2;k++)

            {

                sum=sum+mat1[i][k]\*mat2[k][j];

            }

            mult[i][j]=sum;

        }

    }

    for(i=0;i<=2;i++) //printing the value after multipolication

    {

        for(j=0;j<=2;j++)

        {

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

        }

        printf("\n");

    }

    getch();

    return 0;

}

1. Linear search