**Program 2: WAP using recursion to print fibonacci series upto ‘n’ terms.**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution:**

#include<stdio.h>

#include<conio.h>

int fibonacci (int n)

{

if((n==1) ||(n==0))

{

return(n);

}

else

{

return(fibonacci(n-1) +fibonacci(n-2));

}

}

int main ()

{

int n, i=0;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf ("Input the number of terms for Fibonacci Series:");

scanf("%d”, &n);

printf ("\nFibonnaci Series is as follows\n");

while(i<n)

{

printf ("%d ",fibonacci(i));

i++;

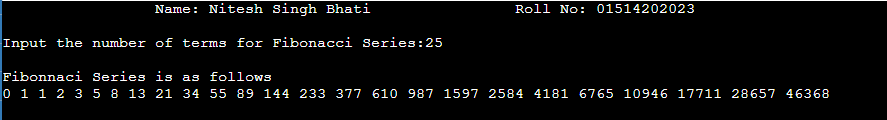
}

getch ();

return 0;

}

**Output:**

****

**Program 3: WAP to find GCD of two numbers using recursion.**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution:**

#include <stdio.h>

#include <conio.h>

int hcf (int n1, int n2);

int main ()

{

printf ("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

int n1, n2;

printf("Enter two positive integers: ");

scanf("%d %d", &n1, &n2);

printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));

return 0;

}

int hcf(int n1, int n2)

{

if (n2 != 0)

{

return hcf(n2, n1 % n2);

}

else

{

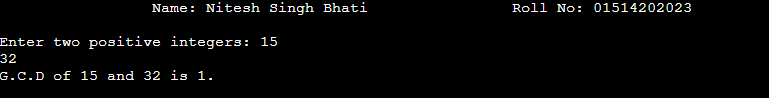
getch ();

return n1;

}

}

**Output:**

****

**Program 4 :WAP to reverse a number using recursion.**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution:**

#include <stdio.h>

#include <conio.h>

int sum=0,rem;

int reverse\_function(int num)

{

if(num)

{

rem=num%10;

sum=sum\*10+rem;

reverse\_function(num/10);

}

else

{

return sum;

}

}

int main()

{

int num,reverse\_number;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter any number:");

scanf("%d",&num);

reverse\_number=reverse\_function(num);

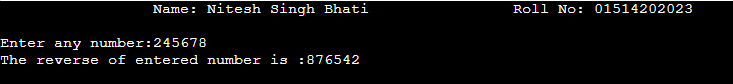
printf("The reverse of entered number is :%d",reverse\_number);

getch ();

return 0;

}

**Output:**



**Program 5 :WAP to implement following operation on one dimensional array (i) Insertion (ii) Deletion (iii) Traversal (iv) Reverse (v) Merge**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution:**

1. #include <stdio.h>

#include <conio.h>

int main() {

int array[100];

int i, n, x, pos;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the number of elements in the array: \n");

scanf("%d", &n);

printf("Enter the elements: \n");

for (i = 0; i < n; i++) {

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

}

printf("Input array elements are: \n");

for (i = 0; i < n; i++) {

printf("%d ", array[i]);

}

printf("\nEnter the new element to be inserted: ");

scanf("%d", &x);

printf("Enter the position where the element is to be inserted: ");

scanf("%d", &pos);

if (pos < 0 || pos > n) {

printf("Invalid position!\n");

return 1; // Exit the program with an error code

}

n = n + 1;

for (i = n - 1; i > pos - 1; i--) {

array[i] = array[i - 1];

}

array[pos - 1] = x;

printf("Array after insertion: \n");

for (i = 0; i < n; i++) {

printf("%d ", array[i]);

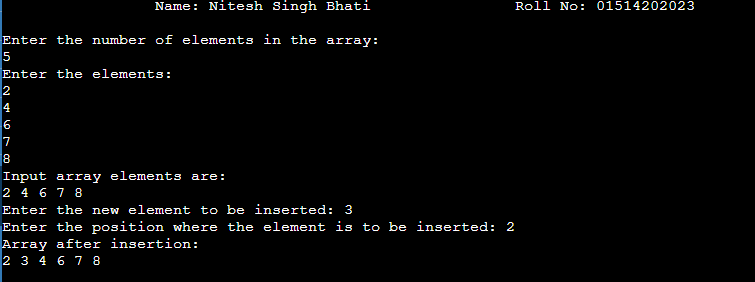
}

getch();

return 0;

}

**Output:**

****

1. #include<stdio.h>

#include<conio.h>

int main()

{

int i,j=0,t,a[10],n,m,s,b[10];

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("\nEnter the Limit:");

scanf("%d",&n);

printf("\nEnter the values:");

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

{

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

}

printf("\nGiven values are:");

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

{

printf("\na[%d]=%d",i,a[i]);

}

printf("\nEnter the position to be deleted:");

scanf("%d",&t);

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

{

if(i!=t)

{

b[j]=a[i];

j++;

}

}

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

{

a[i]=b[i];

}

printf("\nUpdated value is:");

for(i=0;i<n-1;i++)

{

printf("\na[%d]=%d",i,a[i]);

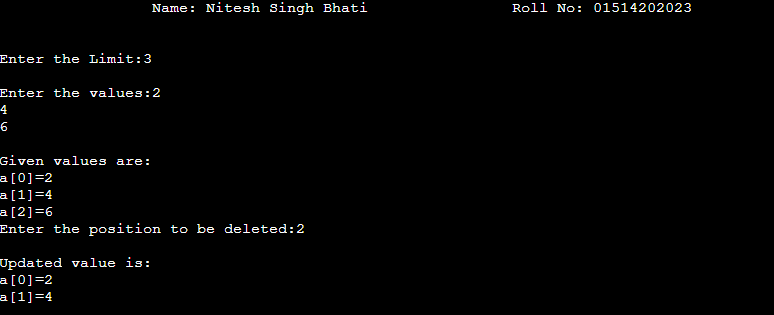
}

getch();

return 0;

}

**Output:**



1. #include <stdio.h>

#include <conio.h>

int main()

{

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

int numbers[5] = {1, 2, 3, 4, 5};

for (int i = 0; i < 5; i++)

{

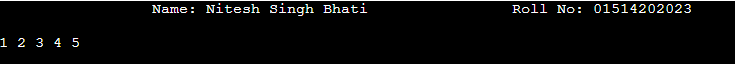
printf("%d ", numbers[i]);

}

return 0;

}

**Output:**



1. #include <stdio.h>

#include <conio.h>

#define N 1000

int main()

{

int arr[N];

int n;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the size of the array: ");

scanf("%d", &n);

printf("Enter an array: ");

for (int i = 0; i< n; i++)

{

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

}

printf("Reversed array: ");

for (int i = n-1; i>=0; i--)

{

printf("%d ", arr[i]);

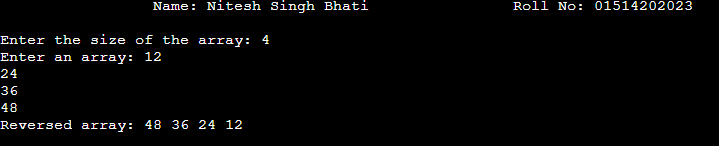
}

getch();

return 0;

}

**Output:**

****

**(v)** #include <stdio.h>

#include<conio.h>

int main() {

int n1, n2, n3;

int a[10000], b[10000], c[20000];

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the size of first array: ");

scanf("%d", &n1);

printf("Enter the array elements: ");

for (int i = 0; i < n1; i++) {

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

}

printf("Enter the size of second array: ");

scanf("%d", &n2);

printf("Enter the array elements: ");

for (int i = 0; i < n2; i++) {

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

}

n3 = n1 + n2;

for (int i = 0; i < n1; i++) {

c[i] = a[i];

}

for (int i = 0; i < n2; i++) {

c[i + n1] = b[i];

}

printf("The merged array: ");

for (int i = 0; i < n3; i++) {

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

}

printf("\nFinal array after sorting: ");

for (int i = 0; i < n3; i++) {

int temp;

for (int j = i + 1; j < n3; j++) {

if (c[i] > c[j]) {

temp = c[i];

c[i] = c[j];

c[j] = temp;

}

}

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

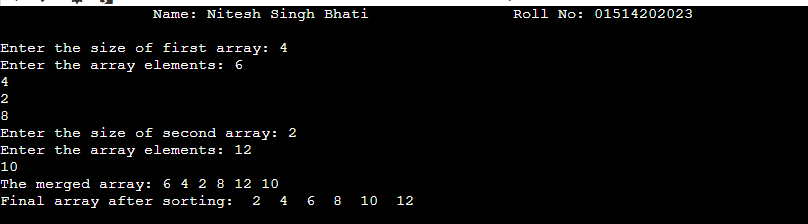
}

getch();

return 0;

}

**Output:**

****

**Program 6 :WAP to search & display the location of an element specified by the user, in an array using (i) Linear Search for unsorted list (ii) Linear Search for sorted list (iii) Binary Search technique using function (iv) Binary Search technique using recursion.**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution:**

1. #include <stdio.h>

#include<conio.h>

int main() {

int array[100], search, c, number;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the number of elements in the array: ");

scanf("%d", &number);

printf("Enter %d numbers\n", number);

for (c = 0; c < number; c++) {

scanf("%d", &array[c]);

}

printf("Enter the number to search: ");

scanf("%d", &search);

for (c = 0; c < number; c++) {

if (array[c] == search) {

printf("%d is present at location %d.\n", search, c + 1);

break;

}

}

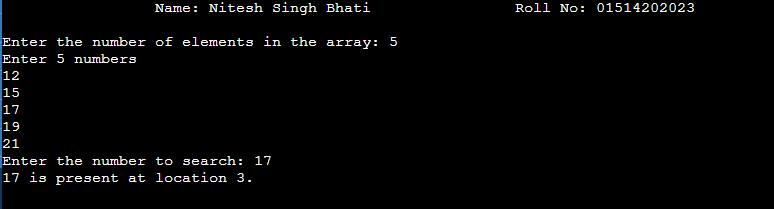
if (c == number) {

printf("%d is not present in the array.\n", search);

}

return 0;}

**Output:**

****

1. #include <stdio.h>

#include<conio.h>

int LinearSearch(int arr[], int n, int num);

int main() {

int arr[50], i, n, num;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter The Size Of The Given Array: ");

scanf("%d", &n);

printf("Enter The Values In Sorted Sequence:\n");

for(i = 0; i < n; i++) {

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

}

printf("Enter The Value To Search Using Linear Search: ");

scanf("%d", &num);

int index = LinearSearch(arr, n, num);

if (index != -1) {

printf("Element %d is found at index: %d\n", num, index);

} else {

printf("Element %d is not found in the list\n", num);

}

return 0;

}

int LinearSearch(int arr[], int n, int num) {

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

if (arr[i] == num) {

return i;

} else if (arr[i] > num) {

return -1;

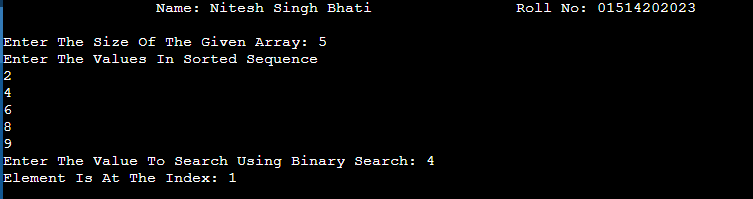
}

}

return -1;

}

**Output:**

****

1. #include <stdio.h>

#include<conio.h>

int binarySearch(int a[], int beg, int end, int val) {

int mid;

if (end >= beg) {

mid = (beg + end) / 2;

if (a[mid] == val) {

return mid + 1;

} else if (a[mid] < val) {

return binarySearch(a, mid + 1, end, val);

} else {

return binarySearch(a, beg, mid - 1, val);

}

}

return -1;

}

int main() {

int a[100], val, n;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the number of elements in the array: ");

scanf("%d", &n);

printf("Enter %d numbers in sorted order: ", n);

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

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

}

printf("Enter the value to search: ");

scanf("%d", &val);

int res = binarySearch(a, 0, n - 1, val);

printf("The elements of the array are - ");

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

printf("%d ", a[i]);

}

if (res == -1) {

printf("\nElement %d is not present in the array.\n", val);

} else {

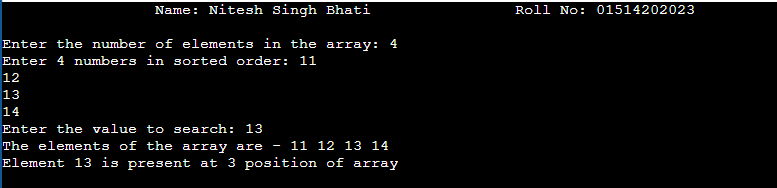
printf("\nElement %d is present at %d position of array\n", val, res);

}

return 0;

}

**Output:**



1. #include<stdio.h>

#include<conio.h>

int main()

{

int c, first, last, middle, n, search, array[100];

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter number of elements\n");

scanf("%d",&n);

printf("Enter %d integers\n", n);

for ( c = 0 ; c < n ; c++ )

scanf("%d",&array[c]);

printf("Enter value to find\n");

scanf("%d",&search);

first = 0;

last = n - 1;

middle = (first+last)/2;

while( first <= last )

{

if ( array[middle] < search )

first = middle + 1;

else if ( array[middle] == search )

{

printf("%d found at location %d.\n", search, middle+1);

break;

}

else

last = middle - 1;

middle = (first + last)/2; }

if ( first > last )

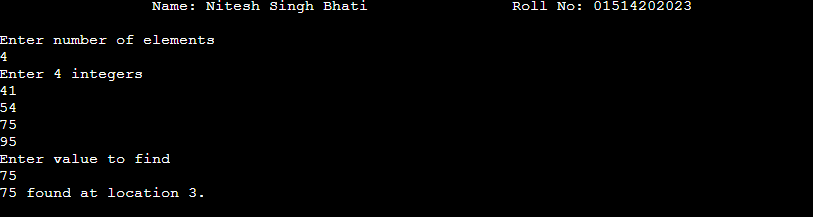
printf("Not found! %d is not present in the list.\n", search);

getch();

return 0;

}

**Output:**

**Program 7:WAP to find the frequency of all elements in a 1-D array and delete the duplicate elements keeping only the first copy of the element.**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution:**

#include <stdio.h>

#include <conio.h>

int main ()

{

int arr[20], i, j, k, size;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf (" Define the number of elements in an array: ");

scanf (" %d", &size);

printf (" \n Enter %d elements of an array: \n ", size);

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

{

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

}

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

{

for ( j = i + 1; j < size; j++)

{

if ( arr[i] == arr[j])

{

for ( k = j; k < size - 1; k++)

{

arr[k] = arr [k + 1];

}

size--;

j--;

}

}

}

printf (" \n Array elements after deletion of the duplicate elements: ");

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

{

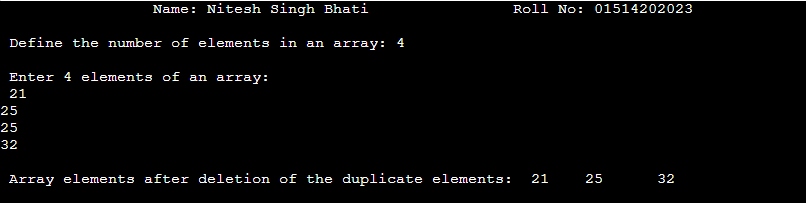
printf (" %d \t", arr[i]);

}

return 0;

}

**Output:**



**Program 8:WAP to Sort an array using menu driven: (i) BUBBLE SORT (ii) SELECTION SORT (iii) INSERTION SORT (iv) MERGE SORT**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution**:

**(i)** #include <stdio.h>

#include <conio.h>

int main()

{

int arr[50], num, x, y, temp;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Please Enter the Number of Elements you want in the array: ");

scanf("%d", &num);

printf("Please Enter the Value of Elements: ");

for(x = 0; x < num; x++)

scanf("%d", &arr[x]);

for(x = 0; x < num - 1; x++)

{

for(y = 0; y < num - x - 1; y++)

{

if(arr[y] > arr[y + 1])

{

temp = arr[y];

arr[y] = arr[y + 1];

arr[y + 1] = temp;

}}

}

printf("Array after implementing bubble sort: ");

for(x = 0; x < num; x++)

{

printf("%d ", arr[x]);

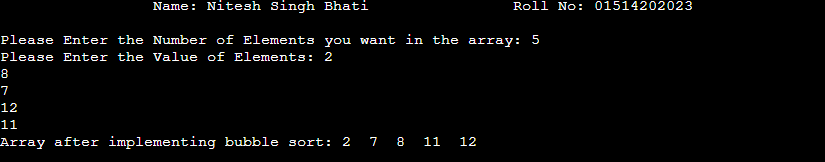
}

getch();

return 0;

}

**Output:**



**(ii)** #include<stdio.h>

#include<conio.h>

void selection(int a[100],int n);

void selection(int a [100],int n)

{

int i,j,temp;

for(i=0;i<=(n-1);i++)

{

for(j=i+1;j<=(n-1);j++)

{

if(a[i]>a[j])

{

temp=a[i];

a[i]=a[j];

a[j]=temp;

}

}

}

}

void main()

{

int a[100],n,i;

clrscr();

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("enter the no of elements:\n");

scanf("%d",&n);

printf("enter elements of the array\n");

for(i=0;i<=(n-1);i++)

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

selection(a,n);

printf("elements of array after selection sort\n");

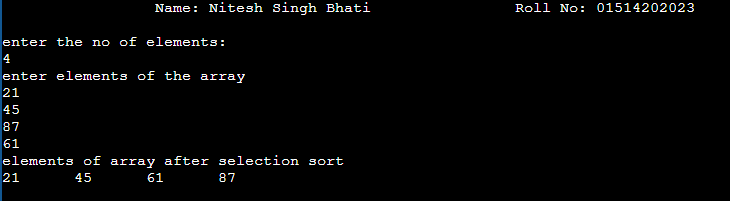
for(i=0;i<=(n-1);i++)

printf("%d\t",a[i]);

getch();

}

**Output:**

**(iii)** #include<stdio.h>

#include<conio.h>

void insertion(int a[100], int n)

{

int i,j,k,temp;

for(i=1;i<=(n-1);i++)

{

for(j=0;j<=(i-1);j++)

{

if(a[j]>a[i])

{

temp=a[j];

a[j]=a[i];

for(k=i;k>j;k--)

{

a[k]=a[k-1];

}

a[k+1]=temp;

}

}

}

}

int main()

{

int a[100],n,i;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("enter number of elements\n");

scanf("%d",&n);

printf("enter element in array\n");

for (i=0;i<=(n-1); i++)

{

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

}

insertion(a,n);

printf("element of array after insertion sort\n");

for (i=0;i<=(n-1);i++)

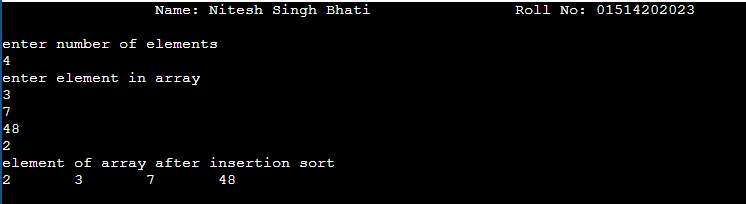
{

printf("%d\t",a[i]);

}

}

**Output:**



1. #include <stdio.h>

#include <conio.h>

void Merge(int arr[], int left, int mid, int right)

{

int i, j, k;

int size1 = mid - left + 1;

int size2 = right - mid;

int Left[size1], Right[size2];

for (i = 0; i < size1; i++)

Left[i] = arr[left + i];

for (j = 0; j < size2; j++)

Right[j] = arr[mid + 1 + j];

i = 0;

j = 0;

k = left;

while (i < size1 && j < size2)

{

if (Left[i] <= Right[j])

{

arr[k] = Left[i];

i++;

}

else

{

arr[k] = Right[j];

j++;

}

k++;

}

while (i < size1)

{

arr[k] = Left[i];

i++;

k++;

}

while (j < size2)

{

arr[k] = Right[j];

j++;

k++;

}

}

void Merge\_Sort(int arr[], int left, int right)

{

if (left < right)

{

int mid = left + (right - left) / 2;

Merge\_Sort(arr, left, mid);

Merge\_Sort(arr, mid + 1, right);

Merge(arr, left, mid, right);

}

}

int main()

{

int size;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the size: ");

scanf("%d", &size);

int arr[size];

printf("Enter the elements of array: ");

for (int i = 0; i < size; i++)

{

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

}

Merge\_Sort(arr, 0, size - 1);

printf("The sorted array is: ");

for (int i = 0; i < size; i++)

{

printf("%d ", arr[i]);

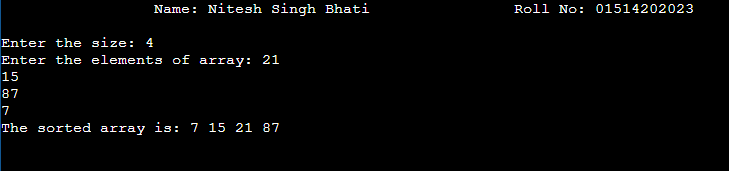
}

printf("\n");

return 0;

}

**Output:**

****

**Program 9:WAP to implement following operation on matrices (i) Addition (ii) Subtraction (iii) Multiplication (iv) Transponse (v) calculate trace of a matrix (vi) calculate norm of a matrix**

**Name : Nitesh Singh Bhati**

**Enroll. No : 01514202023**

**Solution**:

1. #include<stdio.h>

#include<conio.h>

void read(int a[][10], int m, int n);

void traverse(int a[][10], int m, int n);

void mat\_addition(int a[][10], int b[][10], int c[][10], int m, int n);

void read(int a[][10], int m, int n) {

int i, j;

for(i = 0; i < m; i++) {

for(j = 0; j < n; j++) {

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

}

}

}

void traverse(int a[][10], int m, int n) {

int i, j;

for(i = 0; i < m; i++) {

for(j = 0; j < n; j++) {

printf("%d\t", a[i][j]);

}

printf("\n");

}

}

void mat\_addition(int a[][10], int b[][10], int c[][10], int m, int n) {

int i, j;

for(i = 0; i < m; i++) {

for(j = 0; j < n; j++) {

c[i][j] = a[i][j] + b[i][j];

}

}

}

int main() {

int a[10][10], b[10][10], c[10][10], m, n, p, q;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the Order of 1st Matrix: ");

scanf("%d %d", &m, &n);

printf("Enter the Order of 2nd Matrix: ");

scanf("%d %d", &p, &q);

if(m == p && n == q) {

printf("Matrix Addition Is Possible\n\n");

printf("Enter %d Elements of 1st Matrix\n", m \* n);

read(a, m, n);

printf("Enter %d Elements of 2nd Matrix\n", p \* q);

read(b, p, q);

mat\_addition(a, b, c, m, n);

printf("1st Matrix\n");

traverse(a, m, n);

printf("2nd Matrix\n");

traverse(b, p, q);

printf("Resultant Matrix\n");

traverse(c, m, n);

}

else {

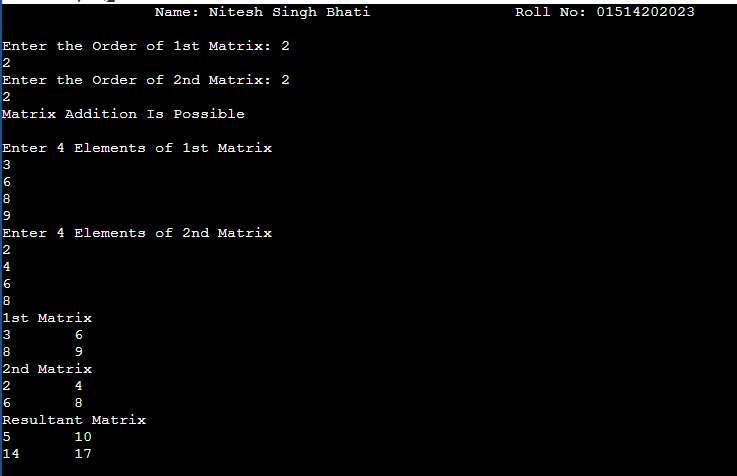
printf("Matrix Addition Is Not Possible\n\n");

}

return 0;

}

**Output:**



**(ii)** #include<stdio.h>

#include<conio.h>

int main()

{

int rowSize1, colSize1, rowSize2, colSize2, i, j;

int mat1[10][10], mat2[10][10], matSub[10][10];

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter Row and Column Size of First Matrix: ");

scanf("%d%d", &rowSize1, &colSize1);

printf("Enter Row and Column Size of Second Matrix: ");

scanf("%d%d", &rowSize2, &colSize2);

if(rowSize1==rowSize2 && colSize1==colSize2)

{

printf("\nEnter First %d\*%d Matrix Elements: ", rowSize1, colSize1);

for(i=0; i<rowSize1; i++)

{

for(j=0; j<colSize1; j++)

scanf("%d", &mat1[i][j]);

}

printf("Enter Second %d\*%d Matrix Elements: ", rowSize2, colSize2);

for(i=0; i<rowSize2; i++)

{

for(j=0; j<colSize2; j++)

scanf("%d", &mat2[i][j]);

}

printf("\nThe Subtraction Result is:\n");

for(i=0; i<rowSize1; i++)

{

for(j=0; j<colSize1; j++)

{

matSub[i][j] = mat1[i][j] - mat2[i][j];

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

}

printf("\n");

}

}

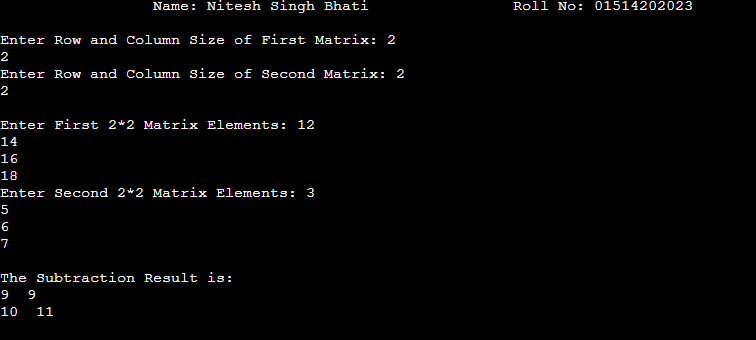
else

printf("\nSubtraction can't be Performed!");

return 0;

}

**Output:**



**(iii)** #include <stdio.h>

#define MAX\_ROWS 10

#define MAX\_COLS 10

int main()

{

int mat1[MAX\_ROWS][MAX\_COLS], mat2[MAX\_ROWS][MAX\_COLS], result[MAX\_ROWS][MAX\_COLS];

int rows1, cols1, rows2, cols2;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter the number of rows and columns of the first matrix: ");

scanf("%d %d", &rows1, &cols1);

printf("Enter the elements of the first matrix:\n");

for(int i=0; i<rows1; i++)

{

for(int j=0; j<cols1; j++)

{

scanf("%d", &mat1[i][j]);

}

}

printf("Enter the number of rows and columns of the second matrix: ");

scanf("%d %d", &rows2, &cols2);

printf("Enter the elements of the second matrix:\n");

for(int i=0; i<rows2; i++)

{

for(int j=0; j<cols2; j++)

{

scanf("%d", &mat2[i][j]);

}

}

if(cols1 != rows2)

{

printf("Matrix multiplication is not possible!\n");

return 0;

}

for(int i=0; i<rows1; i++)

{

for(int j=0; j<cols2; j++)

{

result[i][j] = 0;

for(int k=0; k<cols1; k++)

{

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

}

}

}

printf("Result of matrix multiplication:\n");

for(int i=0; i<rows1; i++)

{

for(int j=0; j<cols2; j++)

{

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

}

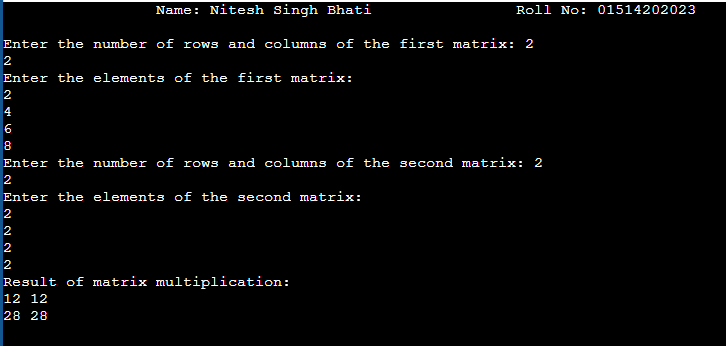
printf("\n");

}

return 0;

}

**Output:**



**(iv)** #include <stdio.h>

#include <conio.h>

int main()

{

int a[10][10], transpose[10][10], r, c;

printf("\t\t Name: Nitesh Singh Bhati \t\t Roll No: 01514202023\n\n");

printf("Enter rows and columns: ");

scanf("%d %d", &r, &c);

printf("\nEnter matrix elements:\n");

for (int i = 0; i < r; ++i)

{

for (int j = 0; j < c; ++j)

{

printf("Enter element a%d%d: ", i + 1, j + 1);

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

}

}

printf("\nEntered matrix: \n");

for (int i = 0; i < r; ++i)

{

for (int j = 0; j < c; ++j)

{

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

if (j == c - 1)

printf("\n");

}

}

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

for (int j = 0; j < c; ++j)

{

transpose[j][i] = a[i][j];

}

}

printf("\nTranspose of the matrix:\n");

for (int i = 0; i < c; ++i)

{

for (int j = 0; j < r; ++j)

{

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

if (j == r - 1)

printf("\n");

}

}

getch();

return 0;

}

**Output:**

