NAME : UDAY SHARMA ROLL NO. 65

SECTION : Q

LAB PRACTICAL

Q1. **C program to perform all arithmetic operations.**

**Input:**

#include<stdio.h>

int main(){

int a , b;

printf("Enter two numbers");

scanf("%d %d",&a,&b);

printf("sum of a and b is: %d\n",a+b);

printf("subtract of a and b is: %d\n",a-b);

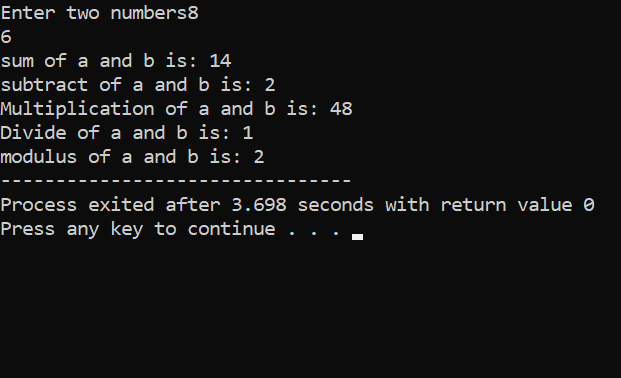
printf("Multiplication of a and b is: %d\n",a\*b);

printf("Divide of a and b is: %d\n",a/b);

printf("modulus of a and b is: %d",a%b);

return 0; }

**OUTPUT:**



**Q2. C program to find area of a triangle if base and height are give**

**Input:**

#include <stdio.h>

int main(){

float base, height, area;

printf("Enter base of the triangle: ");

scanf("%f", &base);

printf("Enter height of the triangle: ");

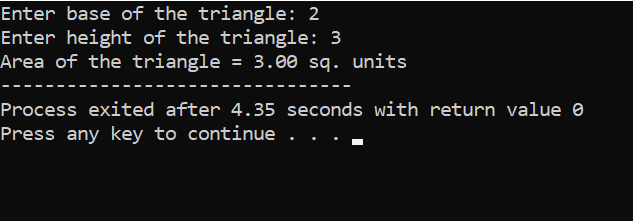
scanf("%f", &height);

area = (base \* height) / 2;

printf("Area of the triangle = %.2f sq. units", area);

return 0;}

**OUTPUT:**



Q3. C program to find all angles of a triangle if two angles are given

**INPUT:**

#include <stdio.h>

int main(){

int a, b, c;

printf("Enter two angles of triangle: ");

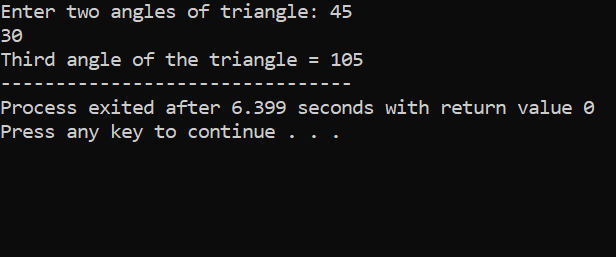
scanf("%d%d", &a, &b);

c = 180 - (a + b);

printf("Third angle of the triangle = %d", c);

return 0;}

**OUTPUT:**



Q4. **C program to convert days in to years, weeks and days.**

**INPUT.**

#include <stdio.h>

int main(){

int days, years, weeks;

printf("Enter days: ");

scanf("%d", &days);

years = (days / 365);

weeks = (days % 365) / 7;

days = days - ((years \* 365) + (weeks \* 7));

printf("YEARS: %d\n", years);

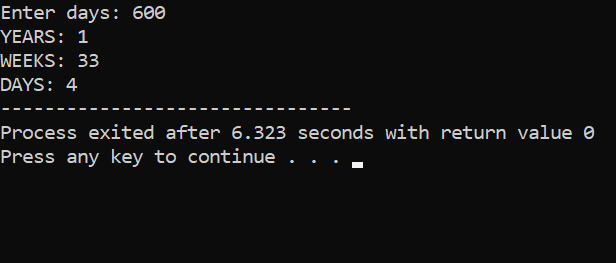
printf("WEEKS: %d\n", weeks);

printf("DAYS: %d", days);

return 0;

}

**OUTPUT:**



**Q5. C program to find power and square root of any number.**

**INPUT:**

#include <stdio.h>

int main()

{

double num, root;

printf("Enter any number to find square root: ");

scanf("%lf", &num);

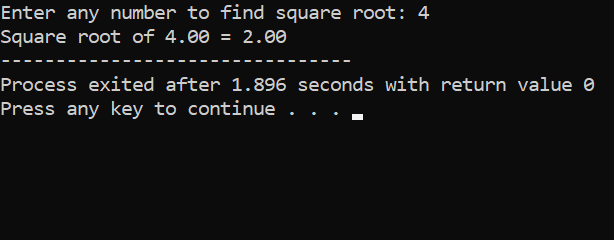
root = sqrt(num);

printf("Square root of %.2lf = %.2lf", num, root);

return 0;

}

**OUTPUT:**



**Q6**. **C program to calculate total, average and percentage and grades of five subjects.**

**Input:**

#include <stdio.h>

int main(){

float eng, phy, chem, math, comp;

float total, average, percentage;

printf("Enter marks of five subjects: :- ");

scanf("%f%f%f%f%f", &eng, &phy, &chem, &math, &comp);

total = eng + phy + chem + math + comp;

average = total / 5.0;

percentage = (total / 500.0) \* 100;

printf("Total marks = %.2f\n", total);

printf("Average marks = %.2f\n", average);

printf("Percentage = %.2f\n", percentage);

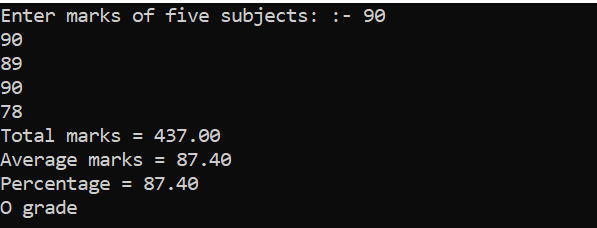
if (percentage<=100)

printf("O grade");

else if(percentage>=80)

printf("A+ grade");}

**output:**



**Q7. C program to check Least Significant Bit (LSB) and MSB of a number using bitwise operator**.

INPUT: LSB PROGRAM

#include <stdio.h>

int main(){

int num;

printf("Enter any number: ");

scanf("%d", &num);

if(num & 1)

printf("LSB of %d is set (1).", num);

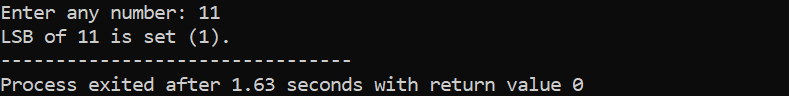
else

printf("LSB of %d is unset (0).", num);

return 0;

}

OUTPUT:



INPUT: MSB PROGRAM

#include <stdio.h>

int main(){

int num;

printf("Enter any number: ");

scanf("%d", &num);

if(num & 1)

printf("LSB of %d is set (1).", num);

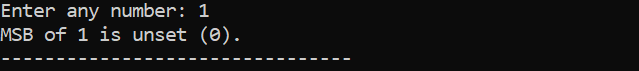
else

printf("LSB of %d is unset (0).", num);

return 0;

}

OUTPUT:



**Q8. C program to swap two numbers USING 3RD VARIABLE AND WITHOUT 3RD VARIABLE.**

**INPUT:**

#include <stdio.h>

int main(){

int var1, var2, temp;

printf("Enter two integrs");

scanf("%d%d", &var1, &var2);

printf("Before SwappingnFirst variable = %d\nSecond variable = %d\n", var1, var2);

temp = var1;

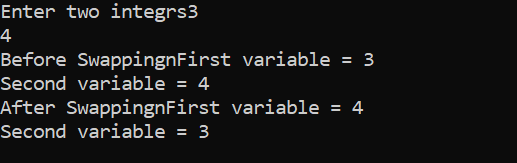
var1 = var2;

var2 = temp;

printf("After SwappingnFirst variable = %d\nSecond variable = %d\n", var1, var2);

return 0;}

**OUTPUT:**



**Q9. C program to find maximum between three numbers using conditional operator AND Ternary Operator.**

**INPUT:**

#include <stdio.h>

int main() {

int a, b, c, max;

printf("Enter Three Integers\n");

scanf("%d %d %d", &a, &b, &c);

max = (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c);

printf("Maximum Number is = %d\n", max);

return 0;

}

**OUTPUT:**



**Q10. C program to check alphabet, digit or special character using Conditional operator.**

**INPUT:**

#include<stdio.h>

int main() {

char ch;

printf("\nEnter Any Character :");

scanf("%c", & ch);

if (ch >= '0' && ch <= '9') {

printf("\n Entered Character is Digit");

} else if (ch >= 'A' && ch <= 'Z') {

printf("\n Entered Character is Capital Letter");

} else if (ch >= 'a' && ch <= 'z') {

printf("\n Entered Character is Small Letter");

} else {

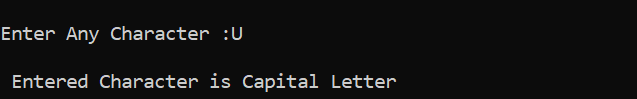
printf("\n Entered Character is Special Character");

}

return 0;

}

**OUTPUT:**



**Q11. C program to calculate total electricity bill**

**INPUT:**

#include<stdio.h>

int main()

{

float bill, units;

printf("Enter the units consumed=");

scanf("%f",&units);

if(units<=50 && units>=0)

{

bill=units\*3.50;

printf("Electricity Bill=%f Rupees",bill);

}

else if(units<=100 && units>50)

{

bill=50\*3.50+(units-50)\*4;

printf("Electricity Bill=%f Rupees",bill);

}

else if(units<=250 && units>150)

{

bill=50\*3.50+100\*4+(units-150)\*5.20;

printf("Electricity Bill=%f Rupees",bill);

}

else if(units>250)

{

bill=50\*3.50+100\*4+100\*5.20+(units-250)\*6.50;

printf("Electricity Bill=%f Rupees",bill);

}

else

{

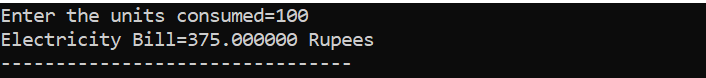
printf("Please enter valid consumed units...");

}

return 0;

}

**OUTPUT:**



**Q12. C program to create Simple Calculator AND Days of week using switch case.**

**INPUT:DAYS OF WEEK BY USING SWITCH**

#include <stdio.h>

int main()

{

int week;

printf("Enter week number(1-7): ");

scanf("%d", &week);

switch(week)

{

case 1:

printf("Monday");

break;

case 2:

printf("Tuesday");

break;

case 3:

printf("Wednesday");

break;

case 4:

printf("Thursday");

break;

case 5:

printf("Friday");

break;

case 6:

printf("Saturday");

break;

case 7:

printf("Sunday");

break;

default:

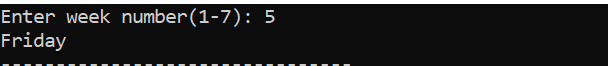
printf("Invalid input! Please enter week number between 1-7.");

}

return 0;

}

**OUTPUT:**



**INPUT : CALCULATOR BY USING SWITCH**

#include <stdio.h>

int main() {

char op;

double first, second;

printf("Enter an operator (+, -, \*, /): ");

scanf("%c", &op);

printf("Enter two operands: ");

scanf("%lf %lf", &first, &second);

switch (op) {

case '+':

printf("%.1lf + %.1lf = %.1lf", first, second, first + second);

break;

case '-':

printf("%.1lf - %.1lf = %.1lf", first, second, first - second);

break;

case '\*':

printf("%.1lf \* %.1lf = %.1lf", first, second, first \* second);

break;

case '/':

printf("%.1lf / %.1lf = %.1lf", first, second, first / second);

break;

default:

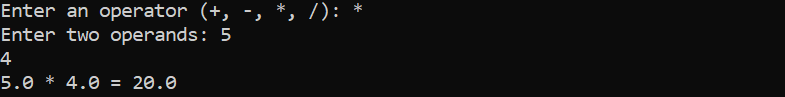
printf("Error! operator is not correct");

}

return 0;

}

**OUTPUT:**



**Q13**. **C program to check vowel or consonant using switch case.**

**INPUT:**

#include <stdio.h>

int main(){

char ch;

printf("Enter any alphabet: ");

scanf("%c", &ch);

switch(ch) {

case 'a':

printf("Vowel");

break;

case 'e':

printf("Vowel");

break;

case 'i':

printf("Vowel");

break;

case 'o':

printf("Vowel");

break;

case 'u':

printf("Vowel");

break;

case 'A':

printf("Vowel");

break;

case 'E':

printf("Vowel");

break;

case 'I':

printf("Vowel");

break;

case 'O':

printf("Vowel");

break;

case 'U':

printf("Vowel");

break;

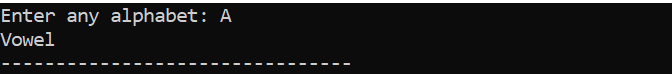
default:

printf("Consonant"); }

return 0;

}

**OUTPUT:**



**Q14. C program to check positive negative or zero using switch case.**

**INPUT:**

#include <stdio.h>

int main(){

int num;

printf("Enter any number: ");

scanf("%d", &num);

switch (num > 0){

case 1:

printf("%d is positive.", num);

break;

case 0:

switch (num < 0) {

case 1:

printf("%d is negative.", num);

break;

case 0:

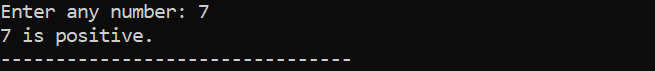
printf("%d is zero.", num);

break;}

break; }

return 0;}

**OUTPUT:**



**Q15.** **C program to check whether a triangle is Equilateral, Isosceles or Scalene.**

**INPUT:**

#include<stdio.h>

int main(){

int side1, side2, side3;

printf("Enter sides of triangle:");

scanf("%d%d%d",&side1,&side2,&side3);

if(side1 == side2 && side2 == side3)

printf("The Given Triangle is equilateral");

else if(side1 == side2 || side2 == side3 || side3 == side1)

printf("The given Triangle is isosceles");

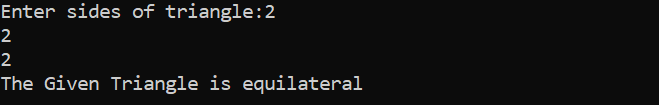
else

printf("The given Triangle is scalene");

return 0;

}

**OUTPUT:**



**Q16. C program to print all natural numbers AND sum of it from 1 to n.**

**INPUT:**

#include <stdio.h>

int main()

{

int i, n;

printf("Enter any number: ");

scanf("%d", &n);

printf("Natural numbers from 1 to %d : \n", n);

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

{

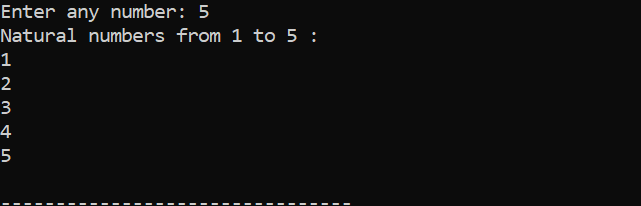
printf("%d\n", i);

}

return 0;

}

**OUTPUT:**



**Q17. C program to print all even numbers AND sum of it from 1 to n.**

**INPUT:**

#include <stdio.h>

int main()

{

int i, n, sum=0;

printf("Enter upper limit: ");

scanf("%d", &n);

for(i=2; i<=n; i+=2)

{

sum += i;

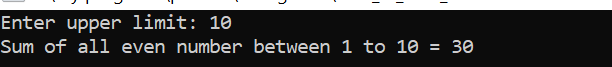
}

printf("Sum of all even number between 1 to %d = %d", n, sum);

return 0;

}

OUTPUT:



**Q18. C program to print multiplication table of a number.**

**INPUT:**

#include <stdio.h>

int main() {

int n, i;

printf("Enter an integer: ");

scanf("%d", &n);

for (i = 1; i <= 10; ++i) {

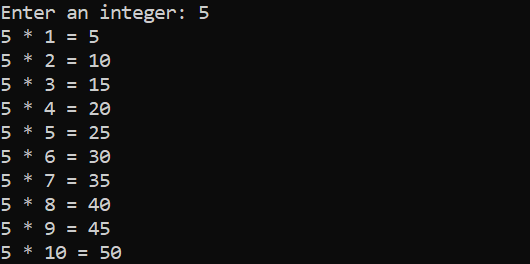
printf("%d \* %d = %d \n", n, i, n \* i);

}

return 0;

}

**OUTPUT:**



**Q19. C program to calculate factorial of a number.**

**INPUT:**

#include<stdio.h>

int main()

{

int fact=1,n;

printf("Enter a number");

scanf("%d",&n);

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

fact=fact\*i;

printf("%d",fact);

}

**OUTPUT:**



**Q20. C program to check whether a number is palindrome or not.**

**INPUT:**

#include<stdio.h>

int main(){

int n,s=0,r,c;

printf("enter a number");

scanf("%d",&n);

c=n;

while(n>0){

r=n%10;

s=r+(s\*10);

n=n/10;}

if(c==s)

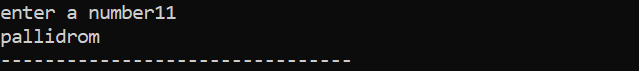
printf("pallidrom");

else

printf("not a palidrom");

return 0;}

OUTPUT:



**Q21. C program to count frequency of digits in a given number.**

**INPUT:**

#include <stdio.h>

#define BASE 10

int main(){

long long num, n;

int i, lastDigit;

int freq[BASE];

printf("Enter any number: ");

scanf("%lld", &num);

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

freq[i] = 0;}

n = num;

while(n != 0 {

lastDigit = n % 10;

n /= 10;

freq[lastDigit]++ }

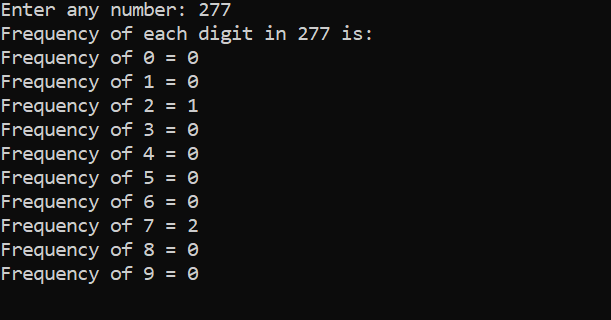
printf("Frequency of each digit in %lld is: \n", num);

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

printf("Frequency of %d = %d\n", i, freq[i]); }

return 0;}

**OUTPUT:**



**Q22. C program to find HCF(GCD) AND LCM of two numbers.**

**INPUT**

#include <stdio.h>

int main() {

int a, b, x, y, t, gcd, lcm;

printf("Enter two integers\n");

scanf("%d%d", &x, &y);

a = x;

b = y;

while (b != 0) {

t = b;

b = a % b;

a = t;}

gcd = a;

lcm = (x\*y)/gcd;

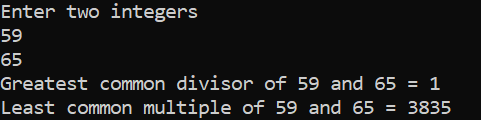
printf("Greatest common divisor of %d and %d = %d\n", x, y, gcd);

printf("Least common multiple of %d and %d = %d\n", x, y, lcm);

return 0;

}

OUTPUT:



**Q23. C program to print all prime numbers between 1 to n.**

**INPUT:**

#include<stdio.h>

void main(){

int i, num, n, count;

printf("Enter the range:");

scanf("%d", &n);

printf("The prime numbers in between the range 1 to %d:",n);

for(num = 1;num<=n;num++){

count = 0;

for(i=2;i<=num/2;i++){

if(num%i==0){

count++;

break;

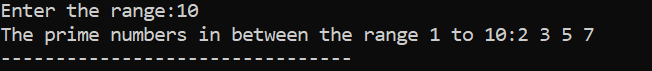
}

}

if(count==0 && num!= 1)

printf("%d ",num); }}

**OUTPUT:**



**Q24. C program to print all Strong Numbers between 1 to n**

**INPUT:**

#include <stdio.h>

int main(){

int i, j, cur, lastDigit, end;

long long fact, sum;

printf("Enter upper limit: ");

scanf("%d", &end);

printf("All Strong numbers between 1 to %d are:\n", end);

for(i=1; i<=end; i++){

cur = i;

sum = 0;

while(cur > 0) {

fact = 1ll;

lastDigit = cur % 10;

for( j=1; j<=lastDigit; j++) {

fact = fact \* j; }

sum += fact;

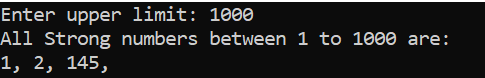
cur /= 10;}

if(sum == i) {

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

return 0;}

**OUTPUT:**



**Q25. C program to print Fibonacci series up to n terms.**

**INPUT**

#include <stdio.h>

int main() {

int i, n;

int t1 = 0, t2 = 1;

int nextTerm = t1 + t2;

printf("Enter the number of terms: ");

scanf("%d", &n);

printf("Fibonacci Series: %d, %d, ", t1, t2);

for (i = 3; i <= n; ++i) {

printf("%d, ", nextTerm);

t1 = t2;

t2 = nextTerm;

nextTerm = t1 + t2;}

return 0;}

**OUTPUT:**



**Q26. C program to print Armstrong numbers from 1 to n AND Check a given number is Armstrong numbers or not.**

**INPUT:**

#include<stdio.h>

int main() {

int n,r,sum=0,temp;

printf("enter the number=");

scanf("%d",&n);

temp=n;

while(n>0) {

r=n%10;

sum=sum+(r\*r\*r);

n=n/10; }

if(temp==sum)

printf("armstrong number ");

else

printf("not armstrong number");

return 0;}

**OUTPUT:**



**Q27. C program to print all Perfect numbers between 1 to n AND Check a given number is Perfect numbers or not.**

**INPUT:**

#include <stdio.h>

int main(){

int i, j, end, sum;

printf("Enter upper limit: ");

scanf("%d", &end);

printf("All Perfect numbers between 1 to %d:\n", end);

for(i=1; i<=end; i++) {

sum = 0;

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

if(i % j == 0) {

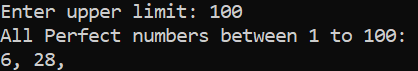
sum += j; } }

if(sum == i) {

printf("%d, ", i);

return 0;}

**OUTPUT:**



**Q28. C program to find power of any number using for loop.**

**INPUT:**

#include <stdio.h>

int main(void) {

int base, exponent, result = 1;

printf("Enter base: ");

scanf("%d", &base);

printf("Enter exponent: ");

scanf("%d", &exponent);

for (int i = 1; i <= exponent; ++i) {

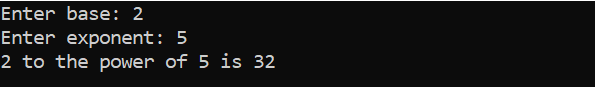
result \*= base;}

printf("%d to the power of %d is %d\n", base, exponent, result);

return 0;

}

**OUTPUT:**



**Q29. C program to print ASCII values of all characters.**

**IUNPUT:**

#include <stdio.h>

int main() {

char c;

printf("Enter a character: ");

scanf("%c", &c);

printf("ASCII value of %c = %d", c, c);

return 0;

}

**OUTPUT:**



**Q30. C program to print Pascal triangle up to n rows.**

**INPUT**

#include<stdio.h>

long factorial(int);

int main() {

int i, n, c;

printf("Enter the number of rows you wish to see in pascal triangle\n");

scanf("%d", & n);

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

for (c = 0; c <= (n - i - 2); c++) printf(" ");

for (c = 0; c <= i; c++) printf("%ld ", factorial(i) / (factorial(c) \* factorial(i - c)));

printf("\n"); }

return 0; }

long factorial(int n) {

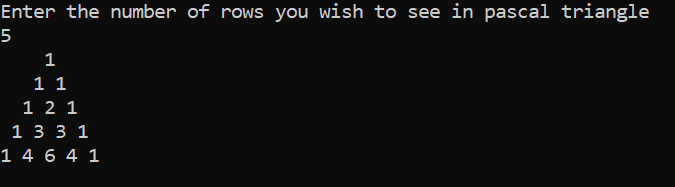
int c;

long result = 1;

for (c = 1; c <= n; c++) result = result \* c;

return result; }

**OUTPUT:**



**Q31. C program to find sum of all elements of array.**

**INPUT:**

#include <stdio.h>

#include <conio.h>

int main(){

int a[1000],i,n,sum=0;

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

scanf("%d",&n);

printf("Enter elements in array : ");

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

{

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

}

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

{ sum+=a[i];

}

printf("sum of array is : %d",sum);

return 0;}

**OUTPUT:**



**Q32. C program to copy one array to another array.**

**INPUT:**

#include <stdio.h>

void main()

{

int arr1[100], arr2[100];

int i, n;

printf("Input the number of elements to be stored in the array :");

scanf("%d",&n);

printf("Input %d elements in the array :\n",n);

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

{

printf("element - %d : ",i);

scanf("%d",&arr1[i]);

}

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

{

arr2[i] = arr1[i];

}

printf("\nThe elements stored in the first array are :\n");

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

{

printf("% 5d", arr1[i]);

}

printf("\n\nThe elements copied into the second array are :\n");

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

{

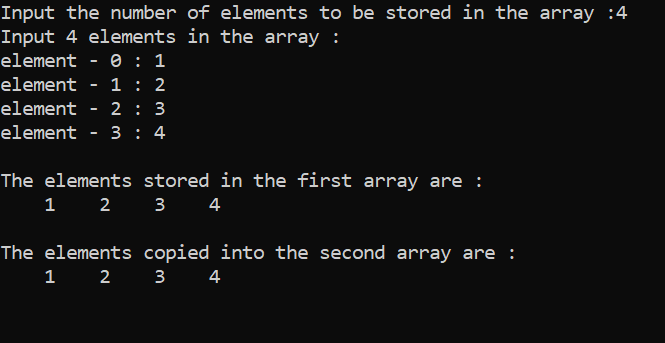
printf("% 5d", arr2[i]);

}

printf("\n\n");

}

**OUTPUT:**



**Q33. C program to insert an element in array at specified position**

**INPUT:**

#include <stdio.h>

int main()

{

int array[100], position, c, n, value;

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

scanf("%d", &n);

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

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

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

printf("Enter the location where you wish to insert an element\n");

scanf("%d", &position);

printf("Enter the value to insert\n");

scanf("%d", &value);

for (c = n - 1; c >= position - 1; c--)

array[c+1] = array[c];

array[position-1] = value;

printf("Resultant array is\n");

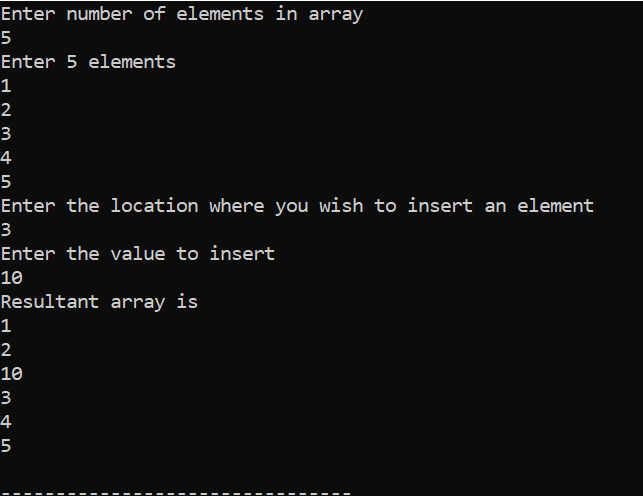
for (c = 0; c <= n; c++)

printf("%d\n", array[c]);

return 0;

}

**OUTPUT:**



**Q34. C program to delete an element in array at specified position**

**INPUT:**

#include <stdio.h>

#define MAX\_SIZE 100

int main()

{

int arr[MAX\_SIZE];

int i, size, pos;

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

scanf("%d", &size);

printf("Enter elements in array : ");

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

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

printf("Enter the element position to delete : ");

scanf("%d", &pos);

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

printf("Invalid position! Please enter position between 1 to %d", size); }

else {

for(i=pos-1; i<size-1; i++){

arr[i] = arr[i + 1]; }

size--;

printf("\nElements of array after delete are : ");

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

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

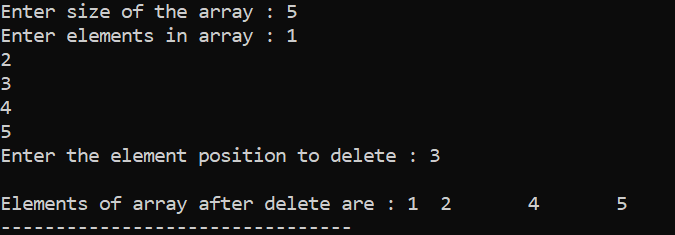
}

}

return 0;

}

**OUTPUT:**



**Q35. C program to search element in array using Linear Search**

**INPUT:**

#include <stdio.h>

int main()

{

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

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

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\n");

scanf("%d",&search);

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

{

if ( array[c] == search ) /\* if required element found \*/

{

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

break;

}

}

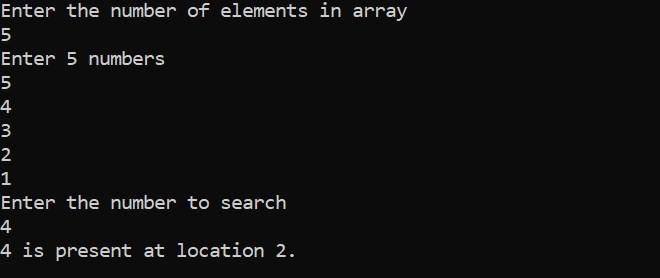
if ( c == number )

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

return 0;

}

**OUTPUT:**



**Q36. C program to find second largest number and Sorting Using Bubble sort in an array**

**INPUT:**

#include <stdio.h>

void main()

{

int a[100],i,j,n,temp;

printf ("Enter the number of elements");

scanf ("%d",&n);

printf("Enter the values");

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

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

}

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

{

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

{

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

{

temp = a[i];

a[i]=a[j];

a[j]=temp;

}

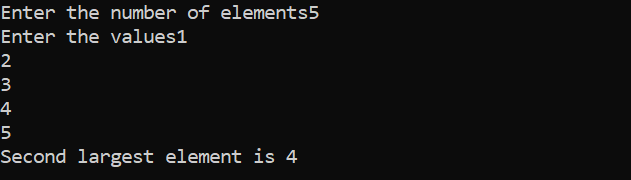
}

}

printf("Second largest element is %d",a[n-2]);

}

OUTPUT:



**Q37. C program to count total number of duplicate elements in an array.**

**INPUT:**

#include <stdio.h>

int main()

{

int arr[10], i, j, Size, Count = 0;

printf("\n Please Enter Number of elements in an array : ");

scanf("%d", &Size);

printf("\n Please Enter %d elements of an Array : ", 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])

{

Count++;

break;}

}

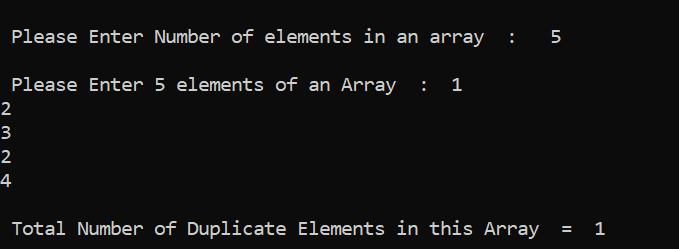
}

printf("\n Total Number of Duplicate Elements in this Array = %d ", Count);

return 0;

}

**OUTPUT:**



Q38. C program to perform scalar matrix multiplication.

INPUT:

#include <stdio.h>

#define SIZE 3

int main(){

int A[SIZE][SIZE];

int num, row, col;

printf("Enter elements in matrix of size %dx%d: \n", SIZE, SIZE);

for(row=0; row<SIZE; row++)

{

for(col=0; col<SIZE; col++)

{

scanf("%d", &A[row][col]);

}

}

printf("Enter any number to multiply with matrix A: ");

scanf("%d", &num);

for(row=0; row<SIZE; row++) {

for(col=0; col<SIZE; col++){

A[row][col] = num \* A[row][col];

}

}

printf("\n Resultant matrix c.A = \n");

for(row=0; row<SIZE; row++) {

for(col=0; col<SIZE; col++){

printf("%d ", A[row][col]); }

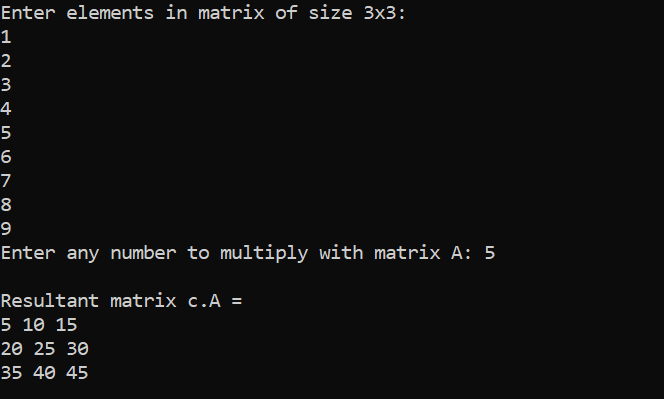
printf("\n");

}

return 0;

}

OUTPUT:



**Q39. C program to find sum of main diagonal elements of a matrix.**

**INPUT:**

#include<stdio.h>

int main()

{

int m,n,i,j,sum=0;

int a[100][100];

printf("enter the size");

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

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

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

printf("enter elements");

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

}

}

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

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

if(i==j){

sum+=a[i][j];

}

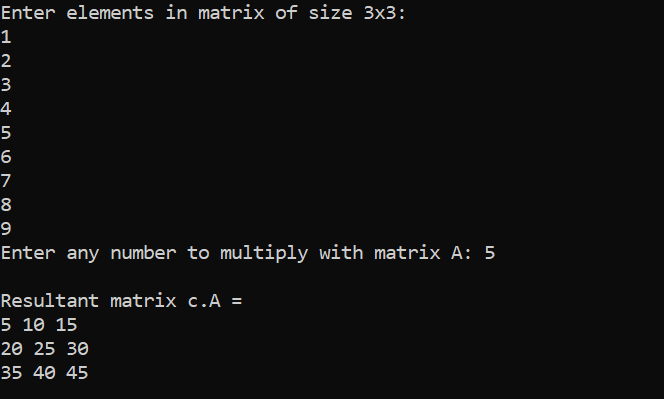
}

}

printf("sum of diagnal elements is %d",sum);

}

**OUTPUT:**



Q40. C program to check sparse AND transpose matrix.

INPUT

#include <stdio.h>

int main() {

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

printf("Enter rows and columns: ");

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

printf("\n Enter 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("\n Entered 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("\n Transpose of the matrix:\n");

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

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

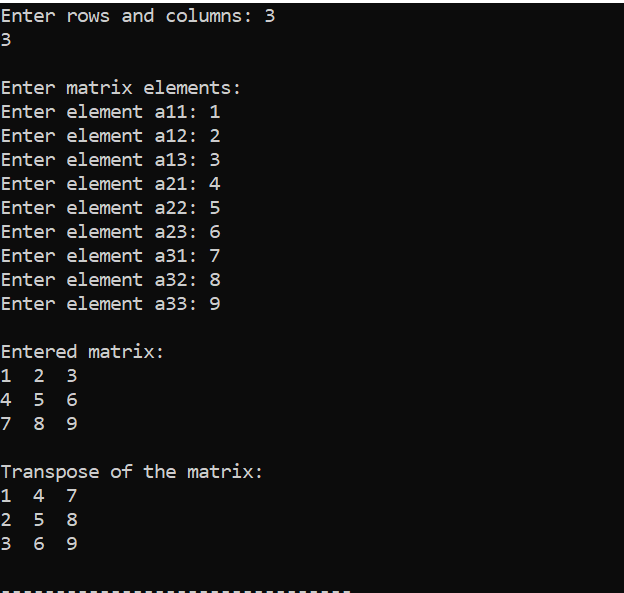
printf("%d ", transpose[i][j]);

if (j == r - 1)

printf("\n"); }

return 0;}

**OUTPUT:**



**Q41. C program to check whether a matrix is Identity matrix or not**

**INPUT:**

#include <stdio.h>

int main(){

int A[SIZE][SIZE];

int row, col, is Identity;

printf("Enter elements in matrix of size 3x3: \n");

for(row=0; row<SIZE; row++) {

for(col=0; col<SIZE; col++){

scanf("%d", &A[row][col]); }

}

Is Identity = 1;

for(row=0; row<SIZE; row++) {

for(col=0; col<SIZE; col++) {

if(row==col && A[row][col]!=1) {

is Identity = 0; }

else if(row!=col && A[row][col]!=0) {

is Identity = 0;}

}

} if(is Identity == 1) {

printf("\n The given matrix is an Identity Matrix.\n");

for(row=0; row<SIZE; row++){

for(col=0; col<SIZE; col++)

{

printf("%d ", A[row][col]);

}

printf("\n"); }

}

else

{

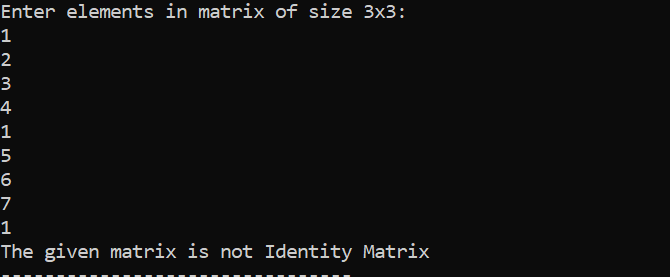
printf("The given matrix is not Identity Matrix");

}

return 0;

}

**OUTPUT:**



**Q42. C program to merge two sorted array in ascending order**

**INPUT:**

#include <stdio.h>

#include <stdlib.h>

int main(void){

int i, n, j, k;

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

scanf("%d", &n);

int arr1[n];

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

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

scanf("%d", &arr1[i]); }

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

scanf("%d", &k);

int arr2[k];

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

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

scanf("%d", &arr2[j]); }

int arr3[n + k];

i = j = 0;

int in;

for (in = 0; in < n + k; in ++) {

if (i < n && j < k) {

if (arr1[i] < arr2[j]) {

arr3[in] = arr1[i];

i++;}

else {

arr3[in] = arr2[j];

j++; } } else if (i < n) {

arr3[in] = arr1[i];

i++; } else {

arr3[in] = arr2[j];

j++;} }

printf("The merged array is: \n");

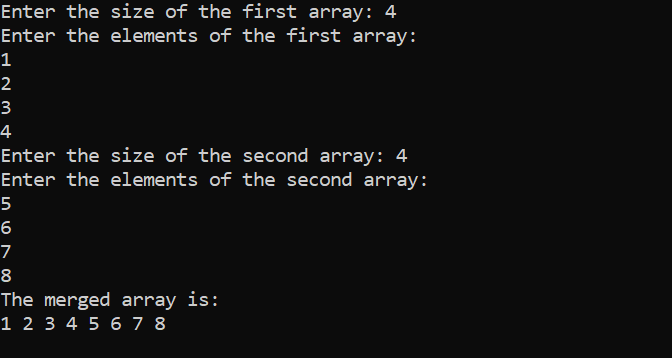
for (in = 0; in < n + k; in++) {

printf("%d ", arr3[in]); }

printf("\n");

return 0;}

**OUTPUT:**



**Q43. C program to check whether a string is palindrome or not without Compare Function of String.**

**INPUT:**

#include <stdio.h>

#include <string.h>

int main(){

char input Array[100], reversed Array[100];

printf("Enter the string for palindrome check \n");

scanf("%s", input Array);

strcpy(reversed Array, input Array);

strrev(reversed Array);

if(strcmp(input Array, reversed Array) == 0 )

printf("%s is a palindrome.\n", input Array);

else

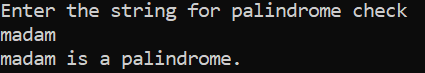
printf("%s is not a palindrome.\n", input Array);

getch();

return 0;

}

OUTPUT:



**Q44. C program to count frequency of each character in a string.**

**INPUT:**

#include<stdio.h>

#include <string.h>

int main()

{

char s[1000];

int i,j,k,count=0,n;

printf("Enter the string : ");

gets(s);

for(j=0;s[j];j++);

n=j;

printf(" frequency count character in string:\n");

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

{

count=1;

if(s[i])

{

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

{

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

{

count++;

s[j]='\0';

}

}

printf(" '%c' = %d \n",s[i],count);

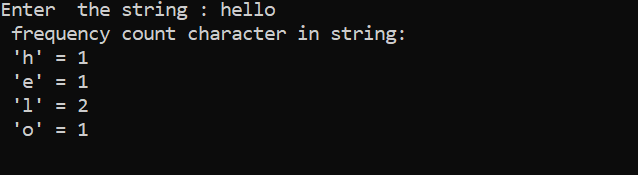
}

}

return0;

}

OUTPUT:



**Q45. C program to find diameter, circumference and area of a circle using functions.**

**INPUT:**

#include <stdio.h>

int main()

{

float radius, diameter, circumference, area;

printf("Enter radius of circle: ");

scanf("%f", &radius);

diameter = 2 \* radius;

circumference = 2 \* 3.14 \* radius;

area = 3.14 \* (radius \* radius);

printf("Diameter of circle = %.2f units \n", diameter);

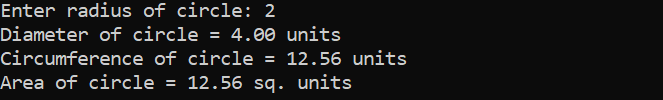
printf("Circumference of circle = %.2f units \n", circumference);

printf("Area of circle = %.2f sq. units ", area);

return 0;

}

**OUTPUT:**



**Q46. C program to check prime, armstrong and perfect numbers using functions.**

**INPUT:**

#include <stdio.h>

int checkPrimeNumber(int n);

int checkArmstrongNumber(int n);

int main() {

int n, flag;

printf("Enter a positive integer: ");

scanf("%d", &n);

flag = checkPrimeNumber(n);

if (flag == 1)

printf("%d is a prime number.\n", n);

else

printf("%d is not a prime number.\n", n);

flag = checkArmstrongNumber(n);

if (flag == 1)

printf("%d is an Armstrong number.", n);

else

printf("%d is not an Armstrong number.", n);

return 0;}

int checkPrimeNumber(int n) {

int i, flag = 1, squareRoot;

squareRoot = sqrt(n);

for (i = 2; i <= squareRoot; ++i) {

if (n % i == 0) {

flag = 0;

break;}}

return flag;}

int check Armstrong Number(int num) {

int original Num, remainder, n = 0, flag;

double result = 0.0;

for (original Num = num; original Num != 0; ++n) {

original Num /= 10;}

for (original Num = num; original Num != 0; original Num /= 10) {

remainder = original Num % 10;

result += pow(remainder, n); }

if (round(result) == num)

flag = 1;

else

flag = 0;

return flag;

}

**OUTPUT:**



**Q47. C program to add two number using pointers.**

**INPUT:**

#include <stdio.h>

int main()

{

int first, second, \*p, \*q, sum;

printf("Enter two integers to add\n");

scanf("%d%d", &first, &second);

p = &first;

q = &second;

sum = \*p + \*q;

printf("Sum of the numbers = %d\n", sum);

return 0;

}

**OUTPUT**



**Q48. Swap 2 numbers using Call by Value AND Call by reference.**

**INPUT:**

#include <stdio.h>

void swap(int, int);

int main(){

int x, y;

printf("Enter the value of x and y\n");

scanf("%d%d",&x,&y);

printf("Before Swapping\nx = %d\ny = %d\n", x, y);

swap(x, y);

printf("After Swapping\nx = %d\ny = %d\n", x, y);

return 0;}void swap(int a, int b){

int temp;

temp = b;

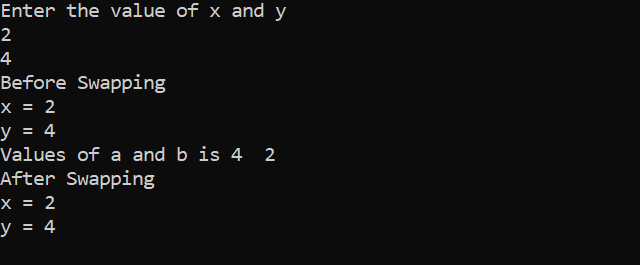
b = a;

a = temp;

printf("Values of a and b is %d %d\n",a,b);

}

**OUTPUT:**



**Q49. C program to copy an array to another array AND reverse an array using pointers.**

**INPUT:**

#include <stdio.h>

#define MAX\_SIZE 100

void printArray(int arr[], int size);

int main()

{

int source\_arr[MAX\_SIZE], dest\_arr[MAX\_SIZE];

int size, i;

int \*source\_ptr = source\_arr;

int \*dest\_ptr = dest\_arr;

int \*end\_ptr;

printf("Enter size of array: ");

scanf("%d", &size);

printf("Enter elements in array: ");

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

{

scanf("%d", (source\_ptr + i));

}

end\_ptr = &source\_arr[size - 1];

printf("\n Source array before copying: ");

printArray(source\_arr, size);

printf("\n Destination array before copying: ");

printArray(dest\_arr, size);

while(source\_ptr <= end\_ptr)

{

\*dest\_ptr = \*source\_ptr;

source\_ptr++;

dest\_ptr++;

}

printf("\n\n Source array after copying: ");

printArray(source\_arr, size);

printf("\n Destination array after copying: ");

printArray(dest\_arr, size);

return 0;

}

void printArray(int \*arr, int size)

{

int i;

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

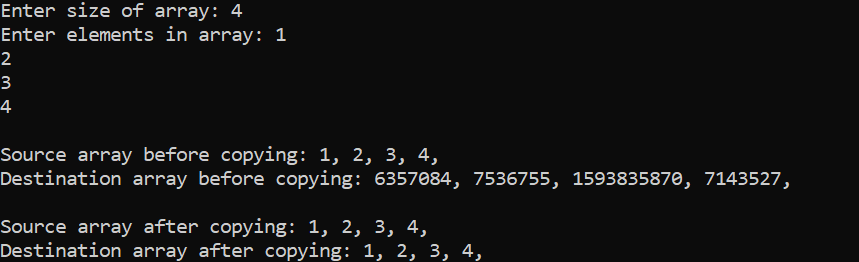
{

printf("%d, ", \*(arr + i));

}

}

**OUTPUT:**



**Q50. All Operations of String.**

**INPUT:**

#include<stdio.h>

#include<conio.h>

void main(){

char string1[25],string2[25];

int l;

clrscr();

Printf("\*\*\*\*\* performing string length \*\*\*\*\*\*\n");

Printf("enter only one string \n");

Scanf("%s",string1);

l = strlen(string1);

printf("the string length is %d\n\n",l);

printf("\*\*\*\* performing string concatenation \*\*\*\*\n");

printf("enter two strings\n");

scanf("%s%s",string1,string2);

printf("the concatenated string is %s\n\n",strcat(string1,string2));

printf("\*\*\*\*\* performing string compare \*\*\*\*\*\n");

printf("enter two strings \n");

scanf("%s%s",string1,string2);

if(strcmp(string1,string2) = = 0)

printf("strings are equal\n");

else

printf("strings are not equal\n");

printf("\*\*\* performing string copy \*\*\*\*\n");

printf("enter the two strings\n");

scanf("%d%d",string1,string2);

printf("the first string is %s and second string is %s\n",string1,string2);

strcpy(string1,string2);

printf("the first string is %s and second string is %s\n",string1,string2);

getch();}

**OUTPUT:**

PATTERNS

**Q1. C PROGRAM TO PRINT PATTERN OF RIGHT ANGLE TRIANGLE.**

**INPUT:**

#include<stdio.h>

int main(){

int n,i;

printf("Enter a number : ");

scanf("%d",&n);

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

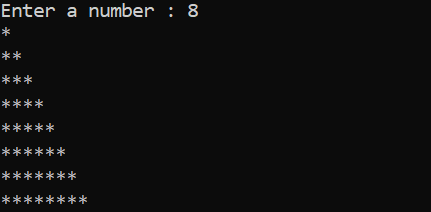
for (int j=1; j<=i; j++){

printf("\*"); }

printf("\n"); }

}

OUTPUT:



Q2. **C PROGRAM TO PRINT PATTERN OF COUNTINGS IN MATRIX FORM.**

**INPUT:**

#include<stdio.h>

int main(){

int i,j,n;

printf("Enter a number : ");

scanf("%d",&n);

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

for (j=1; j<=n; j++) {

printf("%d",i);

}

printf("\n"); }

}

**OUTPUT:**



Q3. **C PROGRAM TO PRINT PATTERN OF COUNTINGS IN MATRIX FORM.**

#include<stdio.h>

int main(){

int i,j,n;

printf("Enter a number : ");

scanf("%d",&n);

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

for (j=1; j<=n; j++) {

printf("%d",j);

}

printf("\n");

}

}

**OUTPUT:**



**Q4.C PROGRAM TO PRINT REVERSE TRIANGLE.**

**INPUT:**

#include<stdio.h>

int main(){

int i,j, rows;

printf("Enter Rows = ") ;

scanf("%d",&rows);

for(i = rows - 1; i > 0; i--) {

for(j = 1; j <= rows - i; j++) {

printf(" "); }

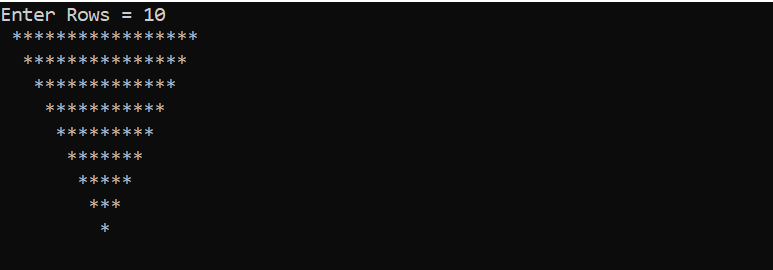
for(j = 1; j <= i \* 2 - 1; j++) {

printf("\*"); }

printf("\n");}

}

**OUTPUT:**



**Q5.C PROGRAM TO PRINT TRIANGLE.**

**INPUT:**

#include<stdio.h>

int main(){

int i, j, rows;

printf("Enter Rows = ");

scanf("%d", &rows);

for(i = 1; i <= rows; i++){

for(j = 1; j <= rows - i; j++){

printf(" "); }

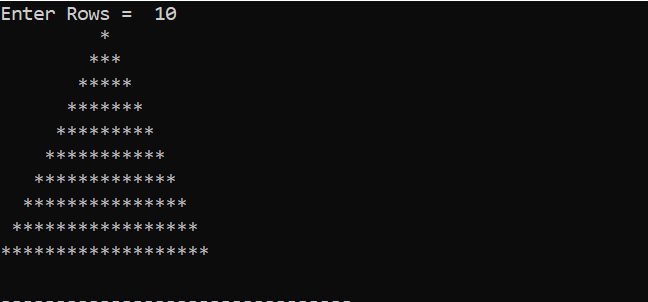
for(j = 1; j <= i \* 2 - 1; j++) {

printf("\*"); }

printf("\n"); }

}

**OUTPUT:**



**Q6.C PROGRAM TO PRINT PATTERN OF DIAMOND.**

**INPUT**:

#include<stdio.h>

int main()

{

int i, j, rows;

printf("Enter Rows = ");

scanf("%d", &rows);

for(i = 1; i <= rows; i++)

{

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

{

printf(" ");

}

for(j = 1; j <= i \* 2 - 1; j++)

{

printf("\*");

}

printf("\n");

}

for (i=rows-1; i>0; i--) {

for (j=1; j<=rows-i; j++) {

printf(" "); }

for (j=1; j<=i\*2-1; j++)

{

printf("\*");

}

printf("\n");

}

}

**OUTPUT:**



**Q7.C PROGRAM TO PRINT PATTERN OF REVERSE RIGHT ANGLE TRIANGLE**

**INPUT:**

#include<stdio.h>

int main(){

int n;

printf("Enter a number : ");

scanf("%d",&n);

for(int i=1; i<=n; i++) {

for (int j=n; j>=i; j--) {

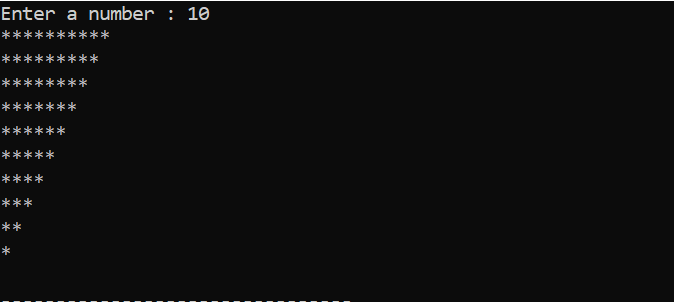
printf("\*") }

printf("\n");

}

}

**OUTPUT:**



**Q8.C PROGRAM TO PRINT THE PATTERN OF TRIANGLE IN NUMERIC FORM.**

**INPUT:**

#include<stdio.h>

int main(){

int i,j,n;

printf("Enter a number : ");

scanf("%d",&n);

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

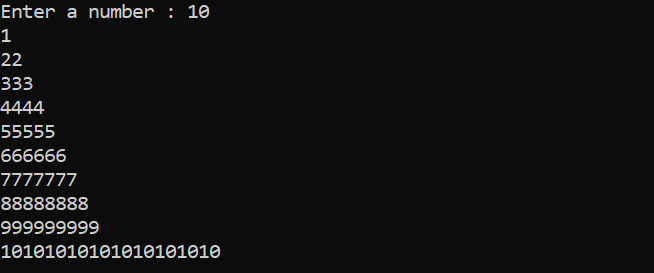
for (j=1; j<=i; j++) {

printf("%d",i); }

printf("\n"); }

}

**OUTPUT:**



**Q9.C PROGRAM TO PRINT PATTERN OF RHOMBUS.**

**INPUT**:

#include <stdio.h>

int main(){

int i, j, n;

printf("Enter Rows : ");

scanf("%d", &n);

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

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

printf(" ");}

for(j=1; j<=n; j++){

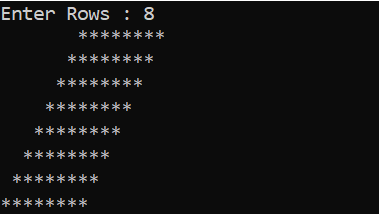
printf("\*");}

printf("\n"); }

return 0;

}

**OUTPUT:**



**Q10.C PROGRAM TO PRINT THE PATTERN OF A LOGO.**

**INPUT:**

#include <stdio.h>

int main(){

int i, j, n;

printf("Enter rows: ");

scanf("%d", &n);

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

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

printf("\*"); }

for(j=i\*2; j<n\*2; j++)

{

printf(" "); }

for(j=i; j>=1; j--)

{

printf("\*");

}

printf("\n");

}

}

**OUTPUT:**

