

Write a program that print the following pattern by using nested loop.

1 2 3 4 5

2 4 6 8

3 6 9

4 8

5

```
#include <stdio.h>
```

```
int main()
```

```
{ int i, j, s;
```

```
for( i=5; i>=1; i--) {
```

```
    for( s=1; s<=5-i; s++) {
```

```
        printf(" ");
```

```
        for( "j=1; j<=i; j++ )
```

```
        { printf("%d", j+s); }
```

```
        printf("\n"); }
```

```
}
```

```
}
```

Write a program that takes 10 elements of array from the user. Calculate and display its ~~minimum~~ maximum element from an array then swap the maximum element with the last of array. Display array before and after swapping

```
#include <stdio.h>
```

```
#include <string.h>
```

```
int main()
```

```
{ int s[10], max, i;
```

```
max = s[0];
```

```
printf("Enter no of Array:");
```

```
for(i=0; i<10; i++)
scanf("%d", &s[i]);
```

```
for(i=0; i<10; i++)
```

```
if (max < s[i])
```

```
{
```

```
max = s[i];
```

```
}
```

```
printf("Value of max before swap = %d", max);
```

```
int temp=0;
```

```
temp = a[10];
```

```
a[10] = max;
```

```
max = temp;
```

```
printf("Max value after swapping = %d", max);
```

```
}
```

Subjective

Write a program that read 10 numbers/elements in an array from user. pass the array to a method that prints the sum of the given number.

```
#include <stdio.h>
int sum of array(int a[]);
int main()
{
    int a[10], i, sum = 0;
    for(i=0; i<10; i++)
    {
        scanf("%d", &a[i]);
    }
    sum = sum of array(a);
    printf("sum of array = %d", sum);
}
int sum of array(int a[])
{
    int i, sum = 0;
    for(i=0; i<10, i++)
    {
        sum = sum + a[i];
    }
    return sum;
}
```

Write down a program that accept integer number from user. Calculate and display its factorial by using recursion function.

```
#include < stdio.h >

int factorial( int n );
int main( )
{
    int n;
    printf (" Enter a number : ");
    scanf ("%d", &n);
    printf ("%d", factorial(n));
}

int factorial( int n )
{
    int fact;
    if (n > 1)
        fact = n * factorial(n - 1);
    return fact;
    else
        fact = 1;
    return fact;
}
```

Write down C program that prints the following

* * * * *
* * * *
* * *
* *
*

```
#include <stdio.h>

int main()
{
    int i, j;
    for( i=1 ; i<=5 ; i++ )
        {
            for( j=i ; j<=5 ; j++ )
                {
                    printf(" * ");
                }
            printf("\n");
        }
}
```

Write down a program that declare a structure student to store rollno(int), name(string) and cgpa(float). The program defines an array to store record of five students. It inputs five students and then display the result of each student.

```
#include < stdio.h >
struct student
{
    int rollno; char name[]; float cgpa; };
int main()
{
    struct student s1, s2, s3, s4, s5;
    printf("Enter roll no of first student:");
    scanf("%d", &s1.rollno);
    printf("Enter name of first student:");
    scanf(" %s", &s1.name);
    printf("Enter cgpa of first student:");
    scanf(" %f", &s1.cgpa);
    printf("Enter detail of second student:");
    printf("Enter name:");
    scanf(" %s", &s2.name);
    printf("Enter roll no:");
    scanf(" %d", &s2.rollno);
    printf("Enter gpa:");
    scanf(" %f", &s2.cgpa);
    printf("Enter detail of third student:");
}
```

```
printf("Enter name:");
scanf("%s", &s3.name);
printf("Enter roll no");
scanf("%d", &s3.rollno);
printf("Enter cgpa");
scanf("%f", &s3.cgpa);
```

```
printf("Result of Student 1:");
printf("%d", s1.rollno);
printf("%s", s1.name);
printf("%f", s1.cgpa);
printf("Result of second student:");
printf("roll no=%d", s2.rollno);
printf("name=%s", s2.name);
printf("cgpa=%f", s2.cgpa);
printf("Result of Third student:");
printf("roll no=%d", s3.rollno);
printf("name=%s", s3.name);
printf("cgpa=%f", s3.cgpa);
printf("Result of forth student:");
printf("roll no=%d", s4.rollno);
printf("name=%s", s4.name);
printf("cgpa=%f", s4.cgpa);
printf("Result of fifth student:");
printf("roll no=%d", s5.rollno);
```

```

printf(" name - %s ", ss.name);
printf(" cgpa - %.2f ", ss.cgpa);

```

3.

Write down a program that prompts a 4 digit number from user and reverse it. Example, if user enters 2348 the program should print 8432.

```

#include <stdio.h>

int main()
{
    int n, c, rev=0;
    printf(" Enter a - digit number: ");
    scanf("%d", &n);
    for (c=n ; c!=0 ; c=c/10)
    {
        rev = 10 * rev + c % 10;
    }
    printf(" Actual no: %d ", n);
    printf(" Reverse no: %d ", rev);
}

```

3.

Write a C program to compute
the following series.

$$G = \frac{1}{1!} + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \dots \text{ upto } n \text{ term.}$$

```
int factorial(int num)
```

```
{ int fact = 1;
```

```
for( int i=1 ; i <= num ; i++ )
```

```
{ fact = fact * i; }
```

```
return fact;
```

```
}
```

```
double calculateSeries( int x, int n )
```

```
{ double sum = 1.0; int fact;
```

```
int power = 3;
```

```
for( int i=2 ; i <= n ; i++ ) {
```

```
fact = factorial(power);
```

```
double term = (-1)^i (x^i) / fact;
```

```
if ( i % 2 == 0 ) {
```

```
sum = sum - term; }
```

```
}
```

```
else { sum = sum + term; }
```

```
power = power + 2;
```

```
}
```

```
return sum;
```

```
}
```

```
int main( ) {
```

```

int x, n;
printf("Enter the value of x:");
scanf("%d", &x);
printf("Enter the number of term:");
scanf("%d", &n);
double result = calculate series(x, n);
printf("The value of the series is:
%lf\n", result);
return 0;
}

```

Write a program that accept two integer value from user and pass their pointers to a function. The function swaps the value of the variable by using their pointers. The swapped valued are displayed by main.

```

void #include <stdio.h>
void swap(int *ptr1, int *ptr2)
{
    int temp = 0;
    temp = *ptr1;
    *ptr1 = *ptr2;
    *ptr2 = temp;
}

```

}

```

int main() {
    int num1, num2;
    printf("Enter two numbers:");
    scanf("%d %d", &num1, &num2);
    swap(&num1, &num2);
    printf("Before swapping: num1 = %d,\n"
           "num2 = %d\n", num1, num2);
    swap(&num1, &num2);
    printf("After swapping: num1 = %d,\n"
           "num2 = %d", num1, num2);
}

```

Write a program that stores 10 values in an array of type integer. The array and its size is passed to the user define function that finds the sum of the all values which are less than the average of all values of array. The sum is displayed by the main function.

```
#include <stdio.h>
```

```

int sum less than average(int arr[], int size)
{
    int sum = 0, average = 0;
    for (int i = 0; i < size; i++) {
        average = average + arr[i];
    }
}

```

}

Sign: _____

```
average = (average / size);  
for (int i = 0; i < size; i++)  
    if (arr[i] < average) {  
        sum = sum + arr[i];  
    }  
return sum;  
}
```

```
int main() :
```

```
{ int size = 10, result;
```

```
int arr[10];
```

```
printf("Enter values of 10 integers:");
```

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

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

```
result = sum less than average (arr, size);
```

```
printf("Sum of values less than
```

```
the average: %d\n", result);
```

```
}
```

Write a program that accept two integer number from user, pass these numbers to user define function as arguments. The function calculate their multiplication by using recursion.

```
#include < stdio.h >
```

```
int multiply (int num1; int num2);
```

```
int main () {
```

```
    int num1, num2, result;
```

```
    printf (" Enter two Integers : ");
```

```
    scanf ("%d %d", &num1, &num2);
```

```
    result = multiply (num1, num2);
```

```
    printf (" Multiplication of %d and %d
```

```
        is = %d \n ", num1, num2, result);
```

```
}
```

```
int multiply (int num1, int num2)
```

```
{    int mul = 0;
```

```
    if (num1 == 0 || num2 == 0)
```

```
        { return 0; }
```

```
else
```

```
    mul = num1 + multiply (num1, num2 - 1);
```

```
    return mul;
```

```
}
```

Date:

Write a program that accept 10 integer values from user in an array and passes array and its size to a function.

The function makes the odd values stored in the array 2-times (i.e. multiply all odd of the array by 2).

The updated array is displayed by the main().

```
void multiply odd by two ( int arr[], int size )
```

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

```
{ if ( arr[i] % 2 != 0 )
```

```
{ arr[i] = arr[i] * 2; }
```

```
int main () { int size = 10 , arr[10];
```

```
printf (" Enter 10 array : " );
```

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

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

```
printf ("\n");
```

```
multiply odd by two ( arr , size );
```

```
printf (" Updated array : \n " );
```

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

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

```
printf ("\n");
```

```
}
```

Write a program that takes year as input (type int) and then determines whether year is leap or not.

```
#include <stdio.h>
int main()
{
    int year;
    printf("Enter year:");
    scanf("%d", &year);
    if (year % 4 == 0)
        printf("Leap year=%d", year);
    else if (year % 100 == 0)
        printf("Not leap year %d", year);
    else if (year % 400 == 0)
        printf("Leap year=%d", year);
    else
        printf("%d leap year", year);
}
```

Write a program that displays the following pattern by using nested loop.

```
*
** *
*** *
**** *
```

```
#include <stdio.h>
```

```
int main()
```

```
{ int i, j, s;
```

Sign:

```

for( i=1 ; i<5 ; i++ ) {
    for( s=1 ; s<=4-i ; s++ )
        printf(" ");
    for( j=1 ; j<=(i+2)-1 ; j++ )
        printf("*");
    printf("\n");
}

```

write a program to find power of a number by using recursion.

```

#include <stdio.h>
int power( int n1, int n2 );
int main() {
    int base, result, n;
    printf("enter base : ");
    scanf("%d", &base);
    printf(" enter number power: ");
    scanf("%d", &n);
    result = power( base, n );
    printf("%d", result);
}

int power( int n1, int n2 ) {
    if( n1 == 0 )
        return ( base * power( base, n-1 ) );
    else
        return 1;
}

```

3

Write a program that takes eight (8) elements (array) from the user. Sort the array in ascending order

```
#include <stdio.h>
```

```
int main() {
```

```
    int a[8], i, temp;
```

```
    printf("enter 8 elements of array: ");
```

```
    for(i=0 ; i<8 ; i++)
```

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

```
}
```

```
    for(i=0 ; i<8 ; i++) {
```

```
        for(j=i+1 ; j<8 ; j++) .
```

```
{
```

```
    if(a[j]<a[i])
```

```
    { temp=a[i];
```

```
        a[i]=a[j];
```

```
        a[j]=temp; } }
```

```

printf(" Sorted Array in Ascending order: ")
for(i=0 ; i<8 ; i++)
{
    printf("%d", a[i]);
}

```

Write a program that defines a structures student to store Std ID (type int), std Roll (type string) and Std Marks (type float).

In main() declare array of structure variable (size 2), input records of two BSCS students and then display the records.

```

#include <stdio.h>
// #include <string.h>
struct student {
    int Std ID;
    char Std Roll[ ];
    float Std Marks;
};

int main()
{
    struct student s1, s2;
    printf(" enter roll no of s1 : ");
    scanf(" %s", s1.Std Roll);

```

```
printf("Enter ID of S1:");
scanf("%d", s1.stdID);
printf("Enter marks of S1:");
scanf("%f", s1.stdMarks);
printf("Enter Roll no. of S2:");
scanf("%s", s2.stdRoll);
printf("Enter ID of S2:");
scanf("%d", s2.stdID);
printf("Enter marks of S2:");
scanf("%f", s2.stdMarks);
printf("Result of first student:");
printf("roll no: %s\nstd-ID = %d\n"
    "Std Marks = %.2f", s1.stdRoll, s1.stdID,
    s1.stdMarks);
printf("Result of second student:");
printf("roll no: %s\nstd-ID = %d\n"
    "Std Marks = %.2f", s2.stdRoll, s2.stdID,
    s2.stdMarks);
```

{

Write a program that accept string from user and pass it to a user defined function named revstring(). The function reverses the string which is then displayed by main().

for Example:

Accept string is TERMINOLOGY

Reversed string is YGOLONIMRET

Note: Don't use any built-in function to reverse the string.

```
#include <stdio.h>
```

```
int revstring(char str[]);
```

```
int main()
```

```
{ int str[100];
```

```
printf("Enter string:");
```

```
scanf("%s", str);
```

```
revstring(str);
```

```
printf("Reverse string = %s", str);
```

```
}
```

```
int revstring(char str[]);
```

```
{ int length = 0;
```

```
char temp;
```

```
while (str[length] != '\0')
```

```
{ length++; }
```

```
int start, end;
```

```

start = 0;
end = length - 1;
if (start < end)
{
    temp = str[start];
    str[start] = str[end];
    str[end] = temp;
}
start += start++;
end--;
}
}

```

Write a program that calculate four function (+, -, *, /) calculator. It asks the user to input two numbers and an operator from the prompt.

```

#include <stdio.h>
int main()
{
    int a, b;
    char op;
    printf("Enter 1st and 2nd number: ");
    scanf("%d %d", &a, &b);
    printf("Enter operator: ");
    scanf("%c", &op);
    switch(op).

```

{

case '+':

printf("%d + %d = %d", a, b, a+b);

break;

case '-':

printf("%d - %d = %d", a, b, a-b);

break;

case '*':

printf("%d * %d = %d", a, b, a*b);

break;

case '/':

printf("%d / %d = %d", a, b, a/b);

break;

default:

printf(" Invalid operator: ");

}

543

54

5

Q # 6: Write a program that accepts a number from user and passes it to a function that determines whether the number is prime or composite. The result is displayed by main function.

```
1 #include <stdio.h>
2
3 int main() {
4     int number;
5     int Composite = 0;
6
7     printf("Enter a positive integer: ");
8     scanf("%d", &number);
9
10    if (number <= 1) {
11        Composite = 1;
12    } else {
13        for (int i = 2; i * i <= number; i++) {
14            if (number % i == 0) {
15                Composite = 1;
16                break;
17            }
18        }
19    }
20 if(Composite)
21 {
22     printf("%d is a composite number.\n", number);
23 } else {
24     printf("%d is not a composite number.\n", number);
25 }
26
27 return 0;
28 }
```



Write a program to calculate the net pay of an employee. Input the basic pay, pass the basic pay to the user defines function PayCalculate(), calculate the net pay and return to main function. Calculate the net pay of an employee.

House rent is 45% of the basic.

Medical allowance is 5% of basic pay if basic pay is greater than Rs.4000/- . It is 7% of basic pay if the basic pay is less or equal than Rs.4000/-.

Conveyance allowance is Rs.91/- If basic pay is less than or equal Rs.4000/-,It is Rs.153/-if the basic pay is more than Rs.4000/-.

Net pay is calculated by adding basic pay, medical allowance, conveyance allowance and house rent.

```
#include <stdio.h>
```

```
int calculate(int pay);
```

```
void main()
```

```
{
```

```
float net = 0;
```

```
int salary = 0;
```

```
printf("enter the basic pay of employ :");
```

```
scanf("%d", &salary);
```

```
net = calculate(salary);
```

```
printf(" net pay=%.2f", net);
```

```
}
```

```
int calculate(int pay)
```

```
{ float total = 0;
```

```
if (pay > 4000)
```

```
{
```

```
total = ((45 * pay / 100) + (5 * pay / 100) + 153 + pay);
```

```
}
```

```
else
```

```
{
```

```
total = ((45 * pay / 100) + (5 * pay / 100) + 1911 + pay);
```

```
}
```

```
return total;
```

```
}
```

5
5 4
5 4 3
5 4 3 2
5 4 3 2 1

```
#include <stdio.h>
int main()
{
    int i, j;
    for(i=5; i>=1; i--)
    {
        for(j=5; j>=i; j--)
        {
            printf("%d", j);
        }
        printf("\n");
    }
}
```

5 4 3 2 1
5 4 3 2
5 4 3
5 4
5

```
#include <stdio.h>
int main()
{
    int i, j;
    for(i=1; i<=5; i++)
    {
```

Date: _____

```
{  
    for(j=5; j>=i; j--)  
        {printf("%d", j);}  
    printf("\n");  
}
```

using nested :

$$\frac{1}{2!} + \frac{2}{3!} + \frac{3}{4!} + \frac{4}{5!} + \frac{5}{6!}$$

include <stdio.h>

int main()

{ int i, j;

float series = 0, f = 1;

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

 for(j=1 ; j<=5 ; j++)

 {
 f = f * (j+1);

Date: _____

Mon Tue Wed Thu Fri

series = series + (j/f); }

printf("sum of series %f", series);

}