

Try yourself: Write a program that converts an uppercase letter to lowercase.

2. Program to compute the quotient and remainder of a division:

```
#include <stdio.h>
int main(){
    int dividend, divisor, quotient, remainder;
    printf("Enter dividend: ");
    scanf("%d", &dividend);
    printf("Enter divisor: ");
    scanf("%d", &divisor);
    quotient=dividend/divisor; //Computes quotient
    remainder=dividend%divisor; //Computes remainder
    printf("Quotient = %d\n", quotient);
    printf("Remainder = %d", remainder);
    return 0;
}
```

Note: For Q2 and Q3, read necessary inputs from user e.g. height and diameter of the cylinder for solving 1(a)

4. Compute quotient and remainder when you divide x by y (read x, y from user). Also, compute quotient and remainder when you divide y by x. Print all these four results.
5. Compute the value of $5x^3 - 4x^2 + 3$; read x from user

Homework Questions:

1. Read a temperature in Celsius and print its Fahrenheit equivalent
2. Print the sum of the series: $1+2+\dots+n$; read n from user
(Hint: $1+2+\dots+n = n(n+1)/2$)
3. Print the sum of the series: $1^2+2^2+\dots+n^2$; read n from user
(Hint: $1^2+2^2+\dots+n^2 = n(n+1)(2n+1)/6$)
4. Read the co-ordinates of two points (x_1, y_1) and (x_2, y_2) from user. Compute the slope (gradient) and print it up to 4 decimal points

Lab Manual 2
CSE 115 (ARA2)

Practice Problems:

1. C program to print the last digit of a number and all the other digits of it:

```
#include<stdio.h>
void main()
{
    int n;
    printf("Enter an integer: ");
    scanf("%d", &n);
    int last = n%10;
    int others = n/10;
    printf("last digit: %d", last);
    printf("\nOther digits: %d", others);
}
```

2. C program to convert days to years weeks and days

```
#include <stdio.h>
void main()
{
    int days, years, weeks;

    // Read total number of days (since the year 0 A.D.) from user
    printf("Enter days: ");
    scanf("%d", &days);

    // Converts days to years, weeks and days
    years = days/365; //Ignoring leap year
    weeks = (days%365)/7;
    days = days- ((years*365) + (weeks*7));

    printf("YEARS: %d\n", years);
    printf("WEEKS: %d\n", weeks);
    printf("DAYS: %d", days);
}
```

C programs to demonstrate bitwise operators:

3. #include <stdio.h>
int main()
{

```

        int i = 3, num=48;
        printf("Right shift by %d: %d\n", i, num>>i);
        printf("\n");
        printf("Left shift by %d: %d\n", i, num<<i);
        return 0;
    }

4. #include <stdio.h>
int main()
{
    int a=12,b=39;
    printf("Output=%d", a&b);
    printf("Output=%d", a|b);
    printf("Output=%d", a^b);
    printf("complement=%d\n", ~35);
    printf("complement=%d\n", ~-12);
    return 0;
}

```

Exercise Problems:

1. Read an integer number n from user. Compute the bitwise AND of n and 1. Do you see a pattern in the result? (Hint: observe the difference in the result when n is odd vs. when n is even) for even output is 0 , odd is 1
2. Read two integer numbers m, n from user. Compute the value of $m \cdot 2^n$ as well as the value of $m \ll n$. Do you see a pattern in the result? Now compute the value of $m/2^n$ as well as the value of $m \gg n$. Do you see a pattern? for left shifted n, multiply by 2^n
right shifted n, divided by 2^n
3. Write a C program to calculate Simple Interest (Formula: Simple Interest = (Principal * Time *Rate) / 100). Read Principal, Time, and Rate from user.
4. Write a C program to calculate Compound Interest. Read the values of P, r, n, and t from user.

The formula for annual compound interest is:

$A = P (1 + r/n)^{nt}$ Where:

A = the future value of the investment/loan, including interest

P = the principal investment amount (the initial deposit or loan amount)

r = the annual interest rate (decimal)

n = the number of times that interest is compounded per year

t = the number of years the money is invested or borrowed for

Assignment Problems:

1. Write a program that calculates the value of $x^2 + \sqrt{x} - 5x$
2. Write a C program to count total number of notes in given amount

Tentative Input/Output (bold ones are user inputs):

Enter amount: **1175**

Total number of notes:

500: 2

100: 1

50: 1

20: 1

10: 0

5: 1

2: 0

1: 0

CSE115L – Computing Concepts Lab

Input two integers and output their sum:

```
#include<stdio.h>
int main()
{
    int var1, var2;
    scanf("%d%d", &var1, &var2);
    printf("%d + %d = %d\n", var1, var2, var1+var2);
    return 0;
}

#include<stdio.h>
int main()
{
    int var1, var2;
    int result;
    scanf("%d%d", &var1, &var2);
    result = var1 + var2;
    printf("%d + %d = %d\n", var1, var2, result);
    return 0;
}
```

Input two double precision floating point numbers and output their sum and division:

```
#include<stdio.h>
int main()
{
    double var1, var2;
    int result1,result2;
    scanf("%lf%lf", &var1, &var2);
    result1 = var1 + var2;
    result2 = var1 / var2;
    printf("%lf + %lf = %d\n", var1, var2, result1);
    printf("%lf / %lf = %d\n", var1, var2, result2);
    return 0;
}

#include<stdio.h>
int main()
{
    double var1, var2;
    double result1,result2;
    scanf("%lf%lf", &var1, &var2);
    result1 = var1 + var2;
    result2 = var1 / var2;
    printf("%lf + %lf = %.2lf\n", var1, var2, result1);
    printf("%lf / %lf = %.2lf\n", var1, var2, result2);
    return 0;
}
```

```

#include<stdio.h>
#include<math.h>

int main()
{
    int x, result;
    printf("Enter a number:");
    scanf("%d", &x);
    result = 2*pow(x, 2)+3*x+1;
    printf("%d", result);
    return 0;
}

#include<stdio.h>
#include<math.h>
int main()
{
    int x;
    double result;
    printf("Enter a number:");
    scanf("%d", &x);
    result = sqrt(3*pow(x, 3)+2*pow(x, 2)+4);
    printf("%lf", result);
    return 0;
}

```

Problems:

1. Write a program to find the average of five numbers. Take all the numbers from user as input.

Sample Output 1:

Enter 5 numbers: 4 8 2 1 5
Average is: 4.00

Sample Output 2:

Enter 5 numbers: 4.1 8.2 2 1.3 5.5
Average is: 4.22

2. Write a program that finds the height and area of a right triangle (90°) using Pythagorean theorem. Take hypotenuse and base as input from the user. Use pow() and sqrt() function.

Sample Output:

Enter base: 3
Enter hypotenuse: 5
Height is: 4.00
Area is: 6.00

$$h = \sqrt{(hp)^2 - (b)^2}$$

3. Write a program to calculate the area and the perimeter of a circle. Take radius as input.

Sample Output:

Enter the radius: 5
Area of the circle: 78.5
Perimeter of the circle: 31.4

```
void main()
{
    int n1, n2;

    printf("Enter any two numbers:\n");
    scanf("%d %d", &n1, &n2);

    // Check if n1 > n2 or not and prints the maximum
    if(n1 > n2)
    {
        printf("%d is maximum", n1);
    }
    else
    {
        printf("%d is maximum", n2);
    }
}
```

3. C program to check Leap Year (a year is a leap year if – (i) it is divisible by 4 but not divisible by 100 OR (ii) it is divisible by 400):

```
#include <stdio.h>

void main()
{
    int year;

    /* Read year from user */
    printf("Enter year : ");
    scanf("%d", &year);

    /* Check for leap year */
    if(((year%4 == 0) && (year%100 !=0)) || (year%400==0))
    {
        printf("LEAP YEAR");
    }
    else
    {
        printf("Not Leap Year");
    }
}
```

EXERCISES:

1. Write a C program to enter month number and print number of days in month
2. Write a C program that decides whether a person is eligible to work in a particular company or not. It does so by following the condition below: If the person's age is between 25 and 45, they are eligible to work. Otherwise, the system says that "You are too young or too old".

ASSIGNMENT PROBLEMS:

1. Write a C program to check whether a number is a multiple of only 5 (e.g. 10), or a multiple of only 11 (e.g. 22), a multiple of both 5 and 11 (e.g. 110), or a multiple of none (e.g. 36).
2. Write a C program to input sides of a triangle and check whether triangle is valid or not (Hint: if sum of any two sides of a triangle is greater than the third side then the triangle is valid)
3. Write a C program to input all angles of a triangle and check whether triangle is valid or not. (Hint: sum of all angles of any triangle is 180 degrees)

CSE 115 Lab on nested if-else, switch-case – Ara2

1. C Program to check whether the input is a character, digit or a special character.

```
#include <stdio.h>

void main()
{
    char ch;
    printf("Enter any character: ");
    scanf("%c", &ch);
    if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
        printf("%c is ALPHABET.\n", ch);
    else if(ch >= '0' && ch <= '9')
        printf("%c is DIGIT.\n", ch);
    else
        printf("%c is SPECIAL CHARACTER.\n", ch);
}
```

Try it yourself 1a: Write a C program to check whether an input alphabet is a vowel or a consonant (assume that the input is an English letter).

Try it yourself 1b: Write a C program to check whether an input alphabet is in uppercase or lowercase (assume that the input is an English letter).

2. C program to check whether an input is an odd number or an even number

```
#include <stdio.h>

void main()
{
    int num;

    printf("Enter any number to check even or odd: ");
    scanf("%d", &num);

    switch(num % 2)
    {
        //If n%2 == 0
        case 0: printf("Number is Even");
                  break;
        //Else if n%2 != 0
        case 1: printf("Number is Odd");
                  break;
    }
}
```

Try it yourself 2a: Write a C program to check whether an input alphabet is a vowel or a consonant using switch case (assume that the input is an English letter).

EXERCISE:

1. Write a C program that takes input the first letter of any of the following fruits name—
Mango ---- Tk.500/kg,
Apple ---- Tk.250/kg,
Banana ----Tk. 130/kg,
Cherry ----Tk. 270/kg, and suggests the price of the fruit as output, using switch case.
2. Find the maximum between two numbers using switch case.
3. Use switch case to make a simple calculator that can add, subtract, multiply or divide two input numbers. The operator should also be taken as an input which is either +, - , * or /

Sample Input/Output:

Enter two numbers: 4.5 2.9

Enter operator: +

$4.5 + 2.9 = 7.4$

Assignment:

1. Write a C program to check whether a year is a leap year or not, using switch case.
2. Farhan must pay a bill to a shopkeeper. The shopkeeper doesn't have any change, so Farhan must have all the notes required to pay the bill. Write a C program that reads the bill and the number of each type of note (500, 100, 50, 20, 10, 5, 2, 1) from user and then output whether it is possible for Farhan to pay the bill or not. If it is possible, then also output the number of each notes required to pay the bill.

Tentative Input/Output (bold ones are user inputs):

Enter amount: **1175**

Enter number of notes:

500: 1

100: 7

50: 1

20: 0

10: 5

5: 1

2: 0

1: 0

Farhan can pay the bill by using the following counts of different notes:

500: 1

100: 6

50: 1

20: 0

10: 2

5: 1

2: 0

1: 0

$\forall > = 7$

$\forall > = 675$

5%. 6000

1175/5000

11.75 500/1175

CSE 115 Lab on simple loop (part 1) – Ara2

1. Write a C program that computes the sum of the series: $1+2+3+\dots+n$, where n is a user input

```
//Program using while loop:  
#include<stdio.h>  
void main()  
{  
    int n, i, sum=0;  
    printf("Enter the value of n:");  
    scanf("%d",&n);  
    i=n;  
    while(i<=n){  
        sum+=i;  
        i++;  
    }  
    printf("sum=%d", sum);  
}
```

```
//Program using for loop:  
#include<stdio.h>  
  
void main()  
{  
    int n, i, sum = 0;  
    printf("Enter the value of n:");  
    scanf("%d",&n);  
  
    for(i=0;i<=n; i++){  
        sum+=i;  
    }  
    printf("sum=%d", sum);  
}
```

Try Yourself 1. Write a C program that computes the sum of the series: $1^2+2^2+\dots+n^2$, where n is a user input
(Hint: replace the statement sum+=i; by the statement sum+=i*i; in any of the above programs).

3. Write a C program that computes the sum of the series: $3+7+11+\dots+n$, where n is a user input

```
#include<stdio.h>  
void main()  
{  
    int n, i, sum = 0;  
    printf("Enter the value of n:");  
    scanf("%d",&n);  
  
    for(i=3;i<=n; i+=4){  
        sum+=i;  
    }  
    printf("sum=%d", sum);  
}
```

4. Write a C program that reads an integer and then computes & prints the factorial of that integer

```
#include<stdio.h>  
void main()  
{  
    int n, i, fact = 1;  
    printf("Enter the value of n:");  
    scanf("%d",&n);  
  
    for(i=1;i<=n; i++){  
        fact*=i;  
    }  
    printf("n!=%d", fact);  
}
```

Problems:

1. Write a program to display all decimal digits (i.e., numbers in the range 0 - 9)
Example output: 0 1 2 3 4 5 6 7 8 9

2. Write a program to display all the letters of the alphabet (a-z). Use two for loops: the 1st one will print all letters in small letters and the 2nd one will print them in capital letters.

Example output: a b c d e z
A B C D Z

3. Write a program to compute the following series using while loop: $4+5+6+\dots+n$

4. Write a program that computes the following series using for loop: $4+7+10+\dots+n$

5. Write a program to compute the following series using while loop: $2^3+3^3+4^3+\dots+n^3$

Assignment Problems:

1. Write a program to compute the following series using while loop: $5^2+9^2+13^2+\dots+n^2$

2. Write a program that takes a minimum number, a maximum number, and common difference and prints the sum of the arithmetic series between them. Example:

Enter minimum: 11

Enter maximum: 19

Enter Common difference: 2

The Sum is: $11 + 13 + 15 + 17 + 19 = 75$

3. Write a C program to print the multiplication table (নামতা) of any given integer number. Example:

Enter number to print the multiplication table of: 5

$5 * 1 = 5$

$5 * 2 = 10$

$5 * 3 = 15$

$5 * 4 = 20$

$5 * 5 = 25$

$5 * 6 = 30$

$5 * 7 = 35$

$5 * 8 = 40$

$5 * 9 = 45$

$5 * 10 = 50$

4. Write a C program to find power of any number using for loop. Don't use pow() function. Example:

Enter base: 2

Enter power: 5

2 to the power 5 = 32

CSE 115 Lab on simple loop (part 2) – Ara2

Try yourself 1: Write a C program to print all odd numbers from 1 to n (n is user input) using for loop.

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

```
void main()
{
    int i, n;

    printf("Print odd numbers till: ");
    scanf("%d", &n); //Read the upper limit
    printf("All odd numbers from 1 to %d are: \n", n);
```

```
for(i=1; i<=n; i++)
{
    if(i%2!=0) // Check if the number is odd
    {
        printf("%d\n", i);
    }
}
```

Try yourself 1: Write a C program to print all even numbers between 100 and 200 using for loop

2. Write a C program to read an integer from user and count the total number of digits in it.

```
#include <stdio.h>

void main()
{
    lint num, count = 0;

    printf("Enter any integer: ");
    scanf("%d", &num);

    while(num != 0)
    {
        count++;
        num /= 10;
    }

    printf("Total digits: %d", count);
}
```

3. Write a C program to read an integer from user and count the total number of nonzero digits in it.

```
#include <stdio.h>

void main()
{
    lint num, count = 0;

    printf("Enter any integer: ");
    scanf("%d", &num);

    while(num != 0)
    {
        //current digit is num%10
        if(num%10 != 0)
            count++;
        num /= 10;
    }

    printf("Total nonzero digits: %d", count);
}
```

Try yourself 2: Write a C program to read an integer and compute the sum of digits in it. Sample Input/Output:

Enter any integer: 452

Sum of digits in 452: 11

3. Write a C program to find all the factors of a number.

```
#include <stdio.h>

void main()
{
    int i, num;

    printf("Enter any number to find its factor: ");
    scanf("%d", &num);

    printf("All factors of %d are: \n", num);
```

```
for(i=1; i<=num/2; i++)
//highest possible factor of num is: num/2
{
    // If num is exactly divisible by i, then i is a factor of num
    if(num%i==0)
    {
        printf("%d\n", i);
    }
}
```

Try yourself 3: Write a C program to print all the odd factors of a given number.

4. Write a C program that can be used to find whether a number is a prime number or not

```
#include <stdio.h>

void main()
{
    int i, n; isPrime = 1; //isPrime is used as a "flag/indicator". Initially we assume
                           //that n is prime and we set isPrime=1 to indicate this. If we
                           //later find that n is not really a prime, then we will set isPrime=0
    printf("Enter any number to check prime: ");
    scanf("%d", &n);
    for(i=2; i<=n/2; i++) //highest possible factor of n is: n/2; so continue as long as i <= n/2
    {
        // If n has a factor other than 1 and itself then it is not prime
        if(n%i==0) //if i is a factor of n (i.e., if n is divisible by i), where i
                   //varies from 2 to n/2, then n is not prime
        {
            isPrime = 0; //here we set isPrime = 0 to indicate that n is not a prime
            break; //go to the first statement after this for loop (break out of loop)
        }
    }

    if(isPrime == 0) // If isPrime == 0 then n is divisible by a value of i;
                     //therefore n cannot be prime
    {
        printf("\n%d is not a prime number", n);
    }
    else // If isPrime==1 then n is not divisible by ANY value of i; so n is a prime no.
    {
        printf("\n%d is a prime number", n);
    }
}
```

Exercise Problems:

1. Write a C program to read an integer from user and output its last and first digit.
2. Write a C program to check whether an input number is a perfect number or not. A perfect number is a positive integer which is equal to the sum of its proper positive factors. For example: 6 is a perfect number; proper divisors/factors of 6 are 1, 2, 3 and $1+2+3 = 6$. Hence 6 is a perfect number.

Assignment Problems:

1. Write a C program to enter any number from user and find the reverse of given number using loop. Sample input/output (**bold ones** are user inputs):
Enter a number: **2345**
Reverse of 2345 is: **5432**
2. Write a C program to read a number from user and check whether given number is a palindrome or not. A number is a palindrome if the number is the same as its reverse for e.g. 23432 is a palindrome but 2345 is not.
3. Write a C program to read any integer from user and compute the reverse of given number. Also output whether the reverse number is prime or not. E.g. Input/Output:
Enter any integer: **4521**
Reverse number is: **1254**. It is not a prime number.
4. Write a C program to compute the sum of digits of an input number and check if this sum is a prime or not.

CSE 115 Lab on simple loop – Ara2

1. Write a C program to read 2 integers: n and r from user and compute the value of $nPr = \frac{n!}{(n-r)!}$

Inefficient Code	Efficient Code
<pre>#include <stdio.h> void main() { int n,r,m=1,d=1,i; printf("Enter n and r: "); scanf("%d %d", &n, &r); //compute n! for(i=1;i<=n;i++) m*=i; //compute (n-r)! for(i=1;i<=n-r;i++) d*=i; printf("nPr = %d", m/d); }</pre>	<pre>#include <stdio.h> void main() { int n,r,p=1,i; printf("Enter n and r: "); scanf("%d %d", &n, &r); //compute nPr = (n-r+1) (n-r+2)... (n-1)n for(i=n-r+1;i<=n;i++) p*=i; printf("nPr = %d", p); }</pre>

Try yourself 1a: Write a C program to read 2 integers: n and r from user and compute the value of $nCr = \frac{n!}{r!(n-r)!}$

2. Write a C program to find Least Common Multiple (LCM) of two given numbers.

```
#include <stdio.h>

void main()
{
    int i, n1, n2, max, lcm=1;

    printf("Enter any two numbers to find LCM: ");
    scanf("%d %d", &n1, &n2);

    i = max = (n1>n2) ? n1 : n2; //compute the max of n1 and n2; this is the lowest possible value of LCM
    while(1) //while condition is always true (1)
    {
        if(i%n1==0 && i%n2==0) // If i is a multiple of both n1 and n2 then i is the LCM of n1 and n2
        {
            lcm = i;
            break; //break out of the loop since LCM has been found
        }
        //we come to the next line if current value of i is not the LCM
        i += max; //values of i are: max, 2*max, 3*max, ..., all of which are possible values of LCM
    }

    printf("\nLCM of %d and %d = %d\n", n1, n2, lcm);
}
```

Try yourself 2a: Write a C program to read 3 integers and compute their LCM.

3. Write a C program display a given number in words starting from its rightmost digit

<pre>#include <stdio.h> void main() { int num, i; printf("Enter any number to print in words: "); scanf("%d", &num); // Finds last digit of the number and print // corresponding digit in words till num becomes 0 while(num!=0) { switch(num%10) { case 0: printf("Zero "); break; case 1: printf("One "); break; } } }</pre>	<pre>case 2: printf("Two "); break; case 3: printf("Three "); break; case 4: printf("Four "); break; case 5: printf("Five "); break; case 6: printf("Six "); break; case 7: printf("Seven "); break; case 8: printf("Eight "); break; case 9: printf("Nine "); break; } num = num/10; }//end of while loop }</pre>
--	--

Exercise Problems:

1. Write a C program to find Greatest Common Divisor (GCD) of two given integers. GCD of two integers is the highest number that totally divides those two integers. E.g. GCD of 15 and 25 is 5.
2. Write a C program to compute the quadruple factorial of a given number $n, q(n) = \frac{(2n)!}{n!}$
3. Write a C program to compute the sum of the series: $1/1! + 1/2! + 1/3! + \dots + 1/n!$ where n is an input.

Assignment Problems:

1. Write a C program to display a given number in words starting from its leftmost digit.
Hint: Compute the reverse of the given number and then use a while loop like practice 3 to print the digits.
E.g., if input number is 1234 your program should print "One Two Three Four".
2. Write a C program to convert a given Binary number to its Decimal equivalent.
3. Write a C program to convert a given decimal number to its binary equivalent.
4. Write a C program to compute the super-factorial of a given number $n, sf(n) = 1! * 2! * 3! * \dots * n!$
5. Write a C program to print small case alphabets from a to z: a, b, ..., z

CSE 115 Lab on nested loop (part 1) – Ara2

C programs to print the following patterns:

1.

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

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

2.

```
#include <stdio.h>  
#define MAX 5  
void main()  
{  
    int i,j;  
    int space=4;  
    for(i=0;i< MAX;i++)  
    {  
        for(j=0;j< space;j++)  
        {  
            printf(" ");  
        }  
        for(j=0;j<=i;j++)  
        {  
            printf("* ");  
        }  
  
        printf("\n");  
        space--;  
    }  
}
```

```
#include <stdio.h>  
void main()  
{  
    int rows, cols, i, j;  
  
    printf("Enter number of rows: ");  
    scanf("%d", &rows);  
  
    for(i=1; i<=rows; i++)  
    {  
        //print no.s  
        for(j=1; j<=i; j++)  
        {  
            printf("%d", j);  
        }  
  
        printf("\n");  
    }  
}
```

Try yourself 1a: Write a C program to print the following pattern of '*'s in 5 rows:

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

3. Write a C program to print the Fibonacci Series upto the 5th term.

```
#include<stdio.h>  
  
void main()  
{  
    int n, first = 0, second = 1, next, c;  
  
    printf("First 5 terms of Fibonacci series are :-\n");
```

```

for ( c = 0 ; c < 5 ; c++ )
{
    if ( c <= 1 )
        next = c;
    else
    {
        next = first + second;
        first = second;
        second = next;
    }
    printf("%d\n",next);
}
}

```

Try Yourself 3: Write program to print the Fibonacci Series upto the n^{th} term , where n is an input.

Exercise Problems:

1. Write a C program to print the following patterns for given number of rows. :

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

* * * * *	*****

* * *	***
	**
*	*

Assignment Problems:

1. Write a C program to print the following two patterns using nested loop:

0
01
010
0101
01010

1
23
456
7890
12345

```

#include <stdio.h>

void main() {
int i, j, n, sum = 0;

printf("Enter any number to print perfect number up to: ");
scanf("%d", &n);

printf("\nAll Perfect numbers between 1 to %d:\n", n);
// Iterates from 1 to n and print if it is perfect number
for(i=1; i<=n; i++) {
sum = 0;

// Checks whether the current number i is Perfect number or not

for(j=1; j<i; j++) {
if(i%j==0) { //if j is a divisor of i then add j with sum
sum += j;
}
}
//now sum = sum of all proper divisors of i
if(sum == i) // If the current number i is Perfect number then print it
printf("%d is Perfect Number\n", i);
}
}//main

```

Try Yourself 1: Write a C program to find sum of perfect numbers between 1 to n.

2. Write a C program to print the Fibonacci Series upto the 5th term.

<pre> #include<stdio.h> void main() { int n, first = 0, second = 1, next, c; for (c = 0 ; c < 5 ; c++) { if (c <= 1) next = c; else { next = first + second; first = second; second = next; } printf("%d\n",next); } </pre>	
---	--

```

#include <stdio.h>

void main()
{
    int i, j, n;

    //Reads number of rows to be printed
    printf("Enter value of n : ");
    scanf("%d", &n);

    for(i=1; i<=n; i++)
    {
        //Prints trailing spaces
        for(j=i; j<n; j++)
        {
            printf(" ");
        }

        //Prints the pyramid pattern
        for(j=1; j<=(2*i-1); j++)
        {
            printf("*");
        }

        printf("\n");
    }
}

```

Try yourself 3: Print hollow pyramid of n lines. $i = n$ || $i + j = n+1$ || $j - i = n - 1$

Exercise:

1. Write a C program using nested loop to print all the composite (*i.e.*, non-prime) numbers from 2 to n where n is a certain number entered by user.
2. Write a C program to print the first n prime numbers where n is an input.
3. Write a C program using nested loop to print the followings for any given input:

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

Assignment:

1. Write a C program to print all palindrome numbers between 1 and n (where n is an input). For e.g. 121 is a palindrome since the reverse of 121 = 121; but 152 is not a palindrome.
2. Write a C program to compute and print the sum of series: $1+(1+2)+(1+2+3)+\dots+(1+2+\dots+n)$.
3. Write a C program where an integer is given as an input. In the output, the number of lines is equal to that number. In every odd numbered line (e.g. 1st, 3rd, 5th, and so on) you have to print the odd numbers from 1 to that line number. And on the even numbered lines, print all the even numbers from 2 to that number.

Sample Input	Output
6	1 2 1 3 2 4 1 3 5 2 4 6

CSE 115 Lab on functions – Ara2

A function that returns the square of a given number:

```
#include <stdio.h>
float square ( float x ); //function declaration
void printSquare ( float x ); //function declaration

int main()
{
    float m, n ;
    printf ( "\nEnter some number for finding square \n");
    scanf ( "%f", &m ) ;
    // function call
    n = computeSquare ( m ) ;
    printf (" \nSquare of the given number %f is %f", m, n );
}

float computeSquare ( float x ) // function definition
{
    float p ;
    p = x * x ;
    return ( p );
}
```

Try yourself 1: Write a C program to print the cube of a number using function.

2. C program to find maximum and minimum between two numbers using functions

```
#include <stdio.h>

/* Function declarations */
int max(int num1, int num2);
int min(int num1, int num2);

void main()
{
    int num1, num2, maximum, minimum;
    printf("Enter any two numbers: ");
    scanf("%d%d", &num1, &num2);
```

```

maximum = max(num1, num2); //Calls the maximum function
minimum = min(num1, num2); //Calls the minimum function

printf("\nMaximum = %d\n", maximum);
printf("Minimum = %d", minimum);
}

// Finds maximum between two numbers.
int max(int num1, int num2)
{
    return (num1 > num2) ? num1 : num2;
}

// Finds minimum between two numbers.
int min(int num1, int num2)
{
    return (num1 > num2) ? num2 : num1;
}

```

Try yourself2: Write C program to find maximum and minimum between three numbers using functions.

Exercise:

1. Write a C program to find diameter, circumference and area of circle using functions for any radius given by the user as the input.
2. Find the sum of the following series using user-defined function: $1^2 + 2^2 + 3^2 + \dots + N^2$

Assignment:

1. Find the sum of the following series using user-defined function:
 (a) $1/1! + 2/2! + 3/3! + \dots + N/N!$ (b) $1/1! + 1/2! + 1/3! + \dots + 1/N!$
2. Write a C code using functions that takes two integers: a and b as inputs and returns the value of a^b .
3. Compute the sum of the following geometric progression using function:

$$1 + r + r^2 + \dots + r^n$$
 (read the values of r and n from user)
4. Write a C program that reads an integer and returns the reverse of that number using function.

CSE 115 Lab on functions and recursion – Ara2

1. C program of a recursive function to find the sum of n natural numbers. For example: the number 5 will give an output of 15 since $5+4+3+2+1 = 15$.

```
#include <stdio.h>
int sum(int n);
void main(){
    int num,add;
    printf("Enter a positive integer:\n");
    scanf("%d",&num);
    add=sum(num);
    printf("sum=%d",add);
}
int sum(int n){
    if(n==0)
        return 0;
    else
        return n+sum(n-1); /*self call to function sum() */
}
```

Try Yourself 1a: Write a C code to find the sum of the following series using RECURSIVE FUNCTIONS:
 $1^2 + 2^2 + 3^2 + \dots + N^2$

2. Code that computes the factorial of a number using recursive function.

```
#include <stdio.h>
int factorial(int x);
void main()
{
    int N,i,result;
    printf("Enter N: \n");
    scanf("%d",&N);
    result = factorial(N);
    printf("%d! = %d",N,result);
}
int factorial(int x)
{
    if (n==1)
        return 1;
    else
        return n*factorial(n-1);
}
```

Try yourself 2a: Write a C program using recursive function to find the inverse product of 1st n natural numbers. E.g. for n= 5, the function should return $(1/5) * (1/4) * (1/3) * (1/2) * (1/1) = 1/120 = 0.00833$

3. C program to print all natural numbers from 1 to n using recursion.

```
#include <stdio.h>
// Recursively prints all natural number between the given range.
void printNaturalNumber(int lowerLimit, int upperLimit)
{
    if(lowerLimit > upperLimit)
        return;

    printf("%d, ", lowerLimit);

    //Recursively calls the function to print next number
    printNaturalNumber(lowerLimit+1, upperLimit);
}

void main()
{
    int limit;

    printf("Print all natural numbers from 1 to : ");
    scanf("%d", &limit);

    printf("All natural numbers from 1 to %d are: ", limit);
    printNaturalNumber(1, limit);
}
```

if ($n > 1$)
 printnat($n-1$)
 printf (" $1/n$ "); n ,
| 2 3 ... n
| 2 3 ... n

Exercise:

1. Write a C code to find the sum of the following series using RECURSIVE FUNCTIONS:
 $1/1! + 2/2! + 3/3! + \dots + N/N!$ $\frac{1}{1} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{(n-1)!}$
2. Compute the value of a^b using recursion, where a and b are given as inputs.
3. Write a C program to print all natural numbers from n to 1 (i.e., from upper to lower) using recursion.

Assignment:

1. Compute the sum of the following geometric progression without recursion:
 $1 + r + r^2 + \dots + r^n$ (read the values of r and n from user)
2. Compute the sum of the above geometric progression using recursion.
3. Write a C program to find sum of digits of a given number using recursion.