1. Write a C program to read a number from user and check whether given number is a square 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. A number is square palindrome if it is both a perfect square and a palindrome. For e.g., 131 is not a square palindrome, but 121 is a square palindrome.
2. Write a C program to find Strong Numbers within *m* and *n* where *m, n* are user inputs. A number is strong if the sum of its digits’ factorials is equal to the number itself. For e.g. 145 is strong because 1!+4!+5! = 145.  
   Sample input/output :  
   Enter starting and ending range: 1 200  
   Strong numbers between 1 and 200 are: 1 2 145
3. Calculate the sum of the following series, where and are provided as user inputs.
4. Write a program in C to print the n-th perfect number. For e.g. 3rd perfect number is 496.
5. Write a C program to convert a binary number into a decimal number.
6. Write a program in C to display the first n Fibonacci numbers. Note that i-th Fibonacci number is the sum of the previous two Fibonacci numbers i.e. it is the sum of (i-1)-th and (i-2)-th Fibonacci numbers. Assume that the 1st two Fibonacci numbers are 0 and 1, respectively. So Fibonacci series is: 0, 1, 2, 3, 5, 8, 13, .....  
   Sample input/output :  
   Enter n: **7**

First 7 Fibonacci numbers: 0, 1, 2, 3, 5, 8, 13,

1. Write separate C programs to print the following patterns for n lines (n is input) using nested loop:

|  |  |  |
| --- | --- | --- |
| **A**  **B C**  **D E F**  **G H I J**  **K L M N O** | 0         01        010        0101       01010 | \*  \* \*  \* \*  \* \*  \* \*  \* \*  \* \*  \* \*  \* |