**Problem #1**

**Print in Binary**

Write a recursive method called *printBitsRecursive* to print an integer in binary. You can assume the integer is 32 bits. Your output should be 32 bits, with a space after every 4 bits. (Note: There should not be any leading space). **No global memory or loops can be used in this task**. Design parameters of your function appropriately and write necessary *main*() function.

|  |  |
| --- | --- |
| **Sample Input(s)** | **Corresponding Output(s)** |
| 0  10  20  512 | 0000 0000 0000 0000 0000 0000 0000 0000  0000 0000 0000 0000 0000 0000 0000 1010  0000 0000 0000 0000 0000 0000 0001 0100  0000 0000 0000 0000 0000 0010 0000 0000 |

**Problem #2**

**Fibonacci Sum**

Using recursion, implement a function that returns the sum of first *n* Fibonacci numbers. Recall that *n*th Fibonacci number *Fn* is defined as follows:

*F1* = 1

*F2* = 1

*Fn* = *Fn-1* + *Fn-2* for *n* > 2

The prototype of the function should be:

***int SumFib(int n)***

You cannot write any other functions except *SumFib* and *main*. **No loops or global variables can be used.**

|  |  |
| --- | --- |
| **Sample Input(s)** | **Corresponding Output(s)** |
| 1  2  3  4  5  6  7 | 1  2  4  7  12  20  33 |