5.1 #20. Let $S_n$ denote the number of $n$-bit strings that do not contain the pattern 00. Show that $S_n = f_{n+1}$, where $f$ denotes the Fibonacci sequence.

Solution. Exercise 19 in the text (the solution of which is in the back of the book) shows that $S_1, S_2, \ldots$ satisfies the recurrence relation $S_n = S_{n-1} + S_{n-2}$ with initial conditions $S_1 = 2$, $S_2 = 3$.

To see then that $S_n = f_{n+1}$ for all $n \geq 1$, we proceed by induction on $n$. The base cases are evidently true, since $S_1 = 2 = f_2$ and $S_2 = 3 = f_3$. So suppose that $S_k = f_{k+1}$ for all $k < n$, where $n \geq 3$; then using the recurrence relations for the Fibonacci sequence and for the sequence $S$, we see that

$$S_{n+1} = S_n + S_{n-1} = f_{n+1} + f_n = f_{n+2}.$$  

This proves the claim by induction.

5.1 #21. By considering the number of $n$-bit strings with exactly $i$ 0’s and Exercise 20, show that

$$f_{n+1} = \sum_{i=0}^{\lfloor (n+1)/2 \rfloor} C(n+1-i,i), \quad n = 1, 2, \ldots$$

Solution. We proved in Exercise 68 of Section 4.2 that the number of $n$-bit strings having exactly $i$ 0’s, with no two 0’s consecutive, is $C(n+i-1, i)$. But $S_n$ is by definition the number of $n$-bit strings without consecutive 0’s, and hence (by the Addition Principle, if you like) $S_n$ is the sum, over the possible values of $i$, of $C(n+i-1, i)$.

So what are the possible values? Observe that in a bit string for which no two 0’s are consecutive, the number of 0’s cannot exceed the number of 1’s by two or more. Therefore we must have $i \leq n - i + 1 \Rightarrow 2i \leq n + 1 \Rightarrow i \leq \lfloor (n+1)/2 \rfloor$ (since $i$ is an integer). We conclude that

$$S_n = \sum_{i=0}^{\lfloor (n+1)/2 \rfloor} C(n+1-i,i).$$

But $S_n = f_{n+1}$ by Exercise 20, so the formula is proved.
5.1 #23. Let $S_n$ denote the number of $n$-bit strings that do not contain the pattern 010. By considering the number of $n$-bit strings that do not contain the pattern 010 that have no leading 0’s (i.e. that begin with 1); that have one leading 0 (i.e. that begin with 01); that have two leading 0’s; and so on, derive the recurrence relation

$$S_n = S_{n-1} + S_{n-3} + S_{n-4} + \ldots + S_1 + 3.$$ 

Solution. Let $\alpha$ be an $n$-bit string which does not contain the pattern 010. Then $\alpha$ falls into exactly one of the following categories:

- $\alpha$ has no leading 0’s;
- $\alpha$ has $k$ leading 0’s for some $1 \leq k \leq n-3$;
- $\alpha$ has $n-2$, $n-1$, or $n$ leading 0’s.

If $\alpha$ is in the first category, then $\alpha$ must begin with a 1, and the remaining substring is an $(n-1)$-bit string which does not contain the pattern 010. There are $S_{n-1}$ such strings, by definition.

If $\alpha$ is in the second category, then $\alpha$ begins with $k$ 0’s, followed by a 1. The next bit must then be a 1, for otherwise $\alpha$ would contain the pattern 010. The remaining substring is an $n-(k+2)$ bit which does not contain the pattern 010, and there are $S_{n-k-2}$ such strings, again by definition.

Finally, there are exactly 3 possible strings in the last category: $0 \cdots 011$, $0 \cdots 001$, and $0 \cdots 000$. Summing over the possibilities, we find that

$$S_n = S_{n-1} + \sum_{k=1}^{n-3} S_{n-k-2} + 3 = S_{n-1} + S_{n-3} + S_{n-4} + \ldots + S_1 + 3,$$

as claimed.

5.1 #24. By replacing $n$ by $n-1$ in the formula of 23, write a formula for $S_{n-1}$. Subtract the formula for $S_{n-1}$ from the formula for $S_n$ and use the result to derive the recurrence relation

$$S_n = 2S_{n-1} - S_{n-2} + S_{n-3}.$$ 

Solution. Substituting $n-1$ for $n$ in the recurrence relation of Exercise 23, we find that

$$S_{n-1} = S_{n-2} + S_{n-4} + S_{n-5} + \ldots + S_2 + S_1 + 3.$$
Clearly the expressions for $S_n$ and $S_{n-1}$ overlap in the sum $S_{n-4} + S_{n-5} + \ldots + S_2 + S_1 + 3$. Discarding this sum from each expression and taking the difference, we find that

$$S_n - S_{n-1} = (S_{n-1} + S_{n-3}) - S_{n-2}.$$ 

Solving for $S_n$ yields $S_n = 2S_{n-1} - S_{n-2} + S_{n-3}$, as desired.

5.1 #56. If $\alpha$ is a bit string, let $C(\alpha)$ be the maximum number of consecutive 0’s in $\alpha$. [Examples: $C(10010) = 2$, $C(00110001) = 3$.] Let $S_n$ be the number of $n$-bit strings $\alpha$ with $C(\alpha) \leq 2$. Develop a recurrence relation for $S_1, S_2, \ldots$

Solution. Observe that $C(\alpha) \leq 2$ if and only if $\alpha$ does not contain the pattern 000; hence we are looking for a recurrence relation for the number of $n$-bit strings which do not contain the pattern 000. We actually did this in class! If you don’t remember how the argument worked, read Example 5.1.6 in the text and repeat this argument (with the obvious modifications) to obtain the relation

$$S_n = S_{n-1} + S_{n-2} + S_{n-3}.$$ 

5.1 #57. The sequence $g_1, g_2, \ldots$ is defined by the recurrence relation

$$g_n = g_{n-1} + g_{n-2} + 1, \quad n \geq 3,$$

and initial conditions $g_1 = 1$, $g_2 = 3$. By using mathematical induction or otherwise, show that

$$g_n = 2f_n - 1, \quad n \geq 1,$$

where $f_1, f_2, \ldots$ denotes the Fibonacci sequence.

Solution. We proceed by induction on $n$. For the base cases $n = 1$ and $n = 2$, we have

$$g_1 = 1 = 2 \cdot 1 - 1 = 2f_1 - 1, \quad g_2 = 3 = 2 \cdot 2 - 1 = 2f_2 - 1.$$

This proves the base cases. For the inductive step, suppose that $g_k = 2f_k - 1$ for all $k < n$, where $n \geq 3$. Then

$$g_n = g_{n-1} + g_{n-2} + 1$$

$$= (2f_{n-1} - 1) + (2f_{n-2} - 1) + 1$$

$$= 2(f_{n-1} + f_{n-2}) - 1 = 2f_n - 1.$$

By induction, the claim is proved.
5.2 #12. Solve the recurrence relation \( a_n = 2na_{n-1} \) given the initial condition \( a_0 = 1 \).

**Solution.** We proceed by iteration:

\[
\begin{align*}
  a_n & = 2na_{n-1} \\
  & = 2n[2(n-1)a_{n-2}] = 2^2n(n-1)a_{n-2} \\
  & = 2^2n(n-1)[2(n-2)a_{n-3}] = 2^3n(n-1)(n-2)a_{n-3} \\
  & = \ldots = 2^nn(n-1)(n-2)\cdots(2)(1)a_0 \\
  & = 2^n(n!)a_0 = 2^n(n!).
\end{align*}
\]

5.2 #16. Solve the recurrence relation \( a_n = 7a_{n-1} - 10a_{n-2} \) given the initial conditions \( a_0 = 5, a_1 = 16 \).

**Solution.** To solve this 2nd order linear homogeneous recurrence relation, we first solve the associated quadratic equation:

\[
t^2 = 7t - 10 \implies t^2 - 7t + 10 = 0 \implies (t-2)(t-5) = 0 \implies t = 2 \text{ or } t = 5.
\]

Then (as shown in class) the sequences \( b_n = 2^n, c_n = 5^n \) both satisfy the same type of recurrence relation; that is, \( b_n = 7b_{n-1} - 10b_{n-2} \) and \( c_n = 7c_{n-1} - 10c_{n-2} \). So \( Bb_n +Cc_n \) satisfies such a recurrence relation as well for any constants \( B \) and \( C \), and hence we look for constants \( B \) and \( C \) with

\[
Bb_0 +Cc_0 = 5, \quad Bb_1+Cc_1 = 16.
\]

Substituting for \( b_0, b_1, c_0, c_1 \), we obtain the equations \( B + C = 5, \ 2B + 5C = 16 \). Solving for \( B \) and \( C \) gives \( B = 3 \) and \( C = 2 \), so

\[
a_n = Bb_n +Cc_n = 3 \cdot 2^n + 2 \cdot 5^n.
\]