Problem #1
1. Give a recursive definition of the sequence \( a_n, n = 1, 2, 3, ... \) if \( a_n = n^2 - n \) (10 points)

Solution:
A recursive definition suggests something of the form
\[
a_n = \begin{cases} 
  c & \text{base case or boundary condition;} \\
  d \ast a_{n-1} + e & x > 1;
\end{cases}
\]
This is a very general form. \( c \) is a constant that is used as a boundary condition, in this case at \( n=1 \). That is, \( a_1 = c \). \( d \) and \( e \) are numbers, possibly 1.

First, a more creative way of looking at the problem; for any \( n > 0 \), we want \( a_n = n^2 - n \), where each call to the recursion will contribute a piece of the overall result, \( n^2 - n \). So first to decompose this value into \( n \) pieces:
\[
n^2 - n = n(n - 1) = \frac{2n(n-1)}{2} = 2 \sum_{i=1}^{n-1} i
\]
\[
2 \sum_{i=1}^{n-1} i = 2(1 + 2 + ... + (n - 1)) = 2 \ast 1 + 2 \ast 2 + ... + 2(n - 1)
\]
This breaks \( n^2 - n \) down into \( n - 1 \) pieces, which is close enough. Note also that at \( n = 1 \), \( n^2 - n = 1 - 1 = 0 \). So \( a_1 = 0 \). So if each \( n \) contributes a piece and \( a_1 \) contributes nothing, that means each non-boundary-condition recursion call contribute a piece. This gives:
\[
a_n = \begin{cases} 
  0 & n = 1 \\
  a_{n-1} + 2(n-1) & n > 1
\end{cases}
\]

Now, a more traditional methodology that will probably be easier to use (but of course, not as entertaining):

\[
a_n = n^2 - n \quad \text{Our given closed form}
\]
\[
a_{n+1} = (n + 1)^2 - (n + 1) \quad \text{Rewrite in } n+1 \text{ form}
\]
\[
= n^2 + 2n + 1 - n - 1 \quad \text{some algebra}
\]
\[
= (n^2 - n) + 2n \quad \text{rewrite to find the } n \text{ form inside } n+1
\]
\[
= a_n + 2n \quad \text{subsitute } a_n \text{ for } n^2 - n \text{ to obtain a recursion: } a \text{ in terms of itself}
\]
\[
a_n = a_{n-1} + 2(n-1) \quad \text{rewrite in } n \text{ form, opposite of that first step}
\]

Problem #2
2. How many 6-character passwords are there starting with "D" or "H" and ending with "3" or "j", assuming each character is an uppercase or a lowercase English letter or a digit form 0 to 9 (10 points)

Solution:
This problem is considering strings of length 6, or:

To determine how many strings there are, analyze what may be placed on each blank. First, there are 26 uppercase letters, 26 lowercase letters, and 10 digits from 0 to 9, given 62 possible elements that can be placed into each blank.
There are also constraints to be considered. First, the string must start with a "D" or an "H", and must end with a "3" or a "j". So now we have the following:

(1) (2) (3) (4) (5) (6)

Here's the analysis summary:

- (1) only "H" or "D", so 2 possibilities
- (2) no restrictions, so 62 possibilities
- (3) no restrictions, so 62 possibilities
- (4) no restrictions, so 62 possibilities
- (5) no restrictions, so 62 possibilities
- (6) only "3" or "j", so 2 possibilities

So we have $2 \times 62 \times 62 \times 62 \times 62 \times 2 = 2^2 \times 62^4$. This is an acceptable final answer.

**Problem #3**

3. How many length-10 bit strings are there that have more than seven 1’s in the strings? (that means at least eight 1’s). (10 points)

More than seven 1’s suggests that we want the distinct sets of bit strings with exactly eight 1’s, exactly nine 1’s, and exactly ten 1’s. Note that these are disjoint subsets of length-10 binary strings.

How many strings of exactly eight 1’s are there? That is the same as how many ways there are to select exactly eight out of ten slots. Thus there are $\binom{10}{8}$ length-10 bit strings with exactly eight 1’s. Similarly for exactly nine and ten 1’s. The final solution looks like:

$$\binom{10}{8} + \binom{10}{9} + \binom{10}{10}$$

which can be written as

$$\sum_{i=8}^{10} \binom{10}{i}$$

**Problem #4**

4. What is the coefficient of $x^5y^7$ in $(2x + y)^{12}$? (10 points)

By the Binomial Theorem on page 327, Theorem 1, from Chapter 4.4 in the Text, there is a way to easily determine the coefficients in $(2x + y)^{12}$.

$$2^5 \times \binom{12}{7}$$

which is trivial to calculate.