Problem 1  For every integer $n \geq 1$, prove by induction that $1^2 + 2^2 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6}$. 
Problem 2 For every integer $n \geq 1$, prove by induction that $1 \times 1! + 2 \times 2! + \cdots + n \times n! = (n + 1)! - 1$.

Recall that $n! = 1 \times 2 \times 3 \times \ldots n$. 
Problem 3 We have $n \times k$ cups. Each of these cups has one of the $k$ different colors. Assume $k$ boxes of size $n$ are available for packing these cups. Prove we can pack these cups in a way that each box has cups of at most two different colors. (Hint: Use induction on $k$.)
Problem 4 Suppose we have a $2^n$ by $2^n$ chessboard with one cell removed. Prove that the board can be covered with L-trominoes.

For example if $n = 2$ and one of the center cells is removed, the tiling is as shown in figure 1.
Problem 5 Consider the following code:

\[
\begin{align*}
    &s \leftarrow 0 \\
    &c \leftarrow 0 \\
    &\text{for } i \leftarrow 1 \text{ to } n \text{ do} \\
    &\quad \text{for } j \leftarrow 2i \text{ to } 2n \text{ do} \\
    &\quad \quad s \leftarrow s + j \\
    &\quad \quad c \leftarrow c + 1 \\
    &\quad \text{end for} \\
    &\text{end for}
\end{align*}
\]

(a) What is \( s \) after running this code?

(b) What is \( c \) after running this code?

(c) What is the running time of the algorithm?
Problem 6 The following code gets $n$ and $c$ as input. What is the running time of the following code?

\[
\begin{align*}
&i \leftarrow c \\
&\textbf{while } i < n \textbf{ do} \\
&\quad i \leftarrow i \times c. \\
&\textbf{end while}
\end{align*}
\]
Problem 7  The following code represents a version of the well-known sorting algorithm "Bubble Sort":

```
for i ← 1 to n do
    for j ← 1 to n - i do
            Swap(A[j], A[j + 1])
        end if
    end for
end for
```

(a) Prove that at the end of the first iteration of the outer for loop, which is indexed by $i$, the largest element of the array would be at its correct position which is the end of the array.

(b) Prove by induction the correctness of the algorithm.

(c) What is the running time of Bubble Sort?
Problem 8 Order the following pairs of functions in terms of order of magnitude. In each case, briefly explain whether $f(n) = O(g(n))$, $f(n) = \Omega(g(n))$, and/or $f(n) = \Theta(g(n))$.

(a) $f(n) = n^3 + 3n + \log(n)^5$, \hspace{1em} $g(n) = 100n^2 + 3n^3 + 10\sqrt{n}$

(b) $f(n) = \sqrt{n} \log(n)$, \hspace{1em} $g(n) = 2\sqrt{n}$

(c) $f(n) = \log(\log(n^2))$, \hspace{1em} $g(n) = \log(\log(3n + 10))$

(d) $f(n) = 3^n$, \hspace{1em} $g(n) = n^2 2^n$

(e) $f(n) = n^{\frac{1}{2}}$, \hspace{1em} $g(n) = n^{\frac{1}{2}} \log(n)$

Problem 9 Prove the following:

(a) $n! = o(n^n)$

(b) $\log(n!) = \Theta(\log(n^n))$. 