1. Let $S$ consist of all positive integers with no prime factor larger than 3, and define:

$$ X = \sum_{k \in S} \frac{1}{k} $$

Thus, the first few terms of the sum are:

$$ X = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{6} + \frac{1}{8} + \frac{1}{9} + \frac{1}{12} + \cdots $$

(a) In the following sum, write a closed-form expression for $k$ in terms of $i$, $j$, and appropriate positive integers, to make it equal to $X$.

$$ X = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \frac{1}{k} $$

(b) Write a closed-form expression for the obtained sum.

2. Prove that $f(n) = \Theta(g(n))$ if and only if $g(n) = \Theta(f(n))$.

3. Solve the following linear recurrence:

$$ f(0) = 0 $$
$$ f(1) = 1 $$
$$ f(n) = 2f(n-1) - f(n-2). $$

4. Use the master theorem to solve the following recurrences.

(a) $T(n) = 9T(n/3) + n.$
(b) $T(n) = 9T(n/3) + n^2.$
(c) $T(n) = 9T(n/3) + n^3.$
5. Consider the sequence defined by \( a_1 = 1, \ a_{n+1} = (n + 1)^2 - a_n \) for \( n \geq 1 \). Find the first six terms. Guess a general formula for \( a_n \) and prove that your answer is correct.

6. Given the following pseudo-code for a recursive algorithm that computes \( n! \), prove by induction that this algorithm is correct.

```plaintext
fact(n) {
    if (n=0) then
        return 1
    else
        return (n * fact(n-1))
}
```