1. (a) Prove that $P(n) = 5n^3 + 50n^2 - 100n$ is $\Theta(n^3)$.

   i. Prove $\Omega(n^3)$.

   
   $$P(n) = 5n^3 + 50n^2 - 100n$$
   
   $$\geq 5n^3 - 100n, \quad n \geq 0 \quad \text{(drop positive terms)}$$
   
   $$\geq 5n^3 - 100n\left(\frac{n}{10}\right)^2, \quad n \geq 10$$
   
   $$\geq 5n^3 - n^3, \quad n \geq 10$$

   ii. Prove $O(n^3)$.

   
   $$P(n) = 5n^3 + 50n^2 - 100n$$
   
   $$\leq 5n^3 + 50n^2, \quad n \geq 0 \quad \text{(drop negative terms)}$$
   
   $$\leq 5n^3 + 50n^2\left(\frac{n}{100}\right) \quad n \geq 100$$
   
   $$\leq 5.5n^3, \quad n \geq 100$$

(b) Consider the following pseudo-code. (The code does not do anything useful except to test you!)

   ```
   x = 0;
   for J = 1 to n
       for K = J+1 to 3*n
           x = x + 1;
   ```

   Let $T(n)$ be the total number of times the innermost statement (increment $x$) is executed. Derive the EXACT value of $T(n)$. Then express the result in $O(\ )$ form.

   
   $$T(n) = \sum_{j=1}^{n} \sum_{k=j+1}^{3n} (1)$$
   
   $$= \sum_{j=1}^{n} (3n - j)$$
   
   $$= 3n^2 - \sum_{j=1}^{n} j$$
   
   $$= 3n^2 - n(n + 1)/2$$
   
   $$= (5n^2 - n)/2$$
   
   $$= \Theta(n^2).$$
2. Use induction to prove that every postage of 13 cents or more can be achieved using only 3-cent stamps and 7-cent stamps. That is, prove that for every integer $n \geq 13$, there exist some non-negative integers $A$ and $B$ such that

$$n = 3A + 7B.$$ 

**Solution:** For the base, $n = 13$,

$$13 = 3 \times 2 + 7 \times 1.$$ 

Next, for any $n \geq 13$, suppose the following hypothesis is true for some non-negative integers $A, B$:

$$n = 3A + 7B. \tag{1}$$

Then, we shall prove that there exist some non-negative integers $A'$ and $B'$ such that

$$n + 1 = 3A' + 7B'.$$

We consider two cases:

(a) $A \geq 2$. Then, from the hypothesis (eq. (1)) it follows that

$$n + 1 = 3(A - 2) + 7(B + 1).$$

(b) $A < 2$, which implies $B \geq 2$. (Why? If both $A$ and $B$ were less than 2, then $n$ would be at most 10. But we know $n \geq 13$.) Therefore, from the hypothesis (eq. (1)), it follows that

$$n + 1 = 3(A + 5) + 7(B - 2).$$
3. Consider the following recurrence relation, where \( n \) is a power of 2.

\[
T(n) \leq \begin{cases} 
0, & n = 1 \\
2T(n/2) + \log n, & n > 1.
\end{cases}
\]

Prove by induction that

\[
T(n) \leq An + B \log n + C
\]

and determine the constants \( A, B, C \).

**Solution:** For the base, \( n = 1 \),

\[
T(1) \leq 0 \leq A.1 + B \log 1 + C.
\]

Thus,

\[
A + C \geq 0.
\]

For \( n \geq 2 \), suppose that

\[
T(m) \leq Am + B \log m + C, \; \forall m < n.
\]

Then,

\[
\begin{align*}
T(n) & \leq 2T(n/2) + \log n, \quad \text{(from the recurrence)} \\
& \leq 2[A(n/2) + B \log(n/2) + C] + \log n \quad \text{(from hypothesis)} \\
& \leq 2[A(n/2) + B \log n - B + C] + \log n \\
& \leq An + (2B + 1) \log n + (2C - 2B) \\
& \leq An + B \log n + C, \quad \text{(needed for the induction step)}
\end{align*}
\]

To satisfy the latter inequality, we pair like-terms (i.e, the \( n \) terms, \( \log n \) terms, and constants) to get:

\[
\begin{align*}
2B + 1 & \leq B \\
2C - 2B & \leq C
\end{align*}
\]

Thus, together with the earlier inequality, we have the three relations:

\[
\begin{align*}
A & \geq -C \\
B & \leq -1 \\
C & \leq 2B.
\end{align*}
\]

Since \( A \) is the constant for the dominant term in the solution, we would like to make \( A \) as small as possible. We observe that the larger we pick the \( B \) value, the larger \( C \) gets, and the smaller \( A \). Thus we pick:

\[
B = -1, C = -2, A = 2.
\]

So we have proved that

\[
T(n) \leq 2n - \log n - 2.
\]
4. Consider a $2^n \times 2^n$ board, with one of its four quadrants missing. That is, the board consists of only three quadrants, each of size $2^{n-1} \times 2^{n-1}$. Let’s call such a board a quad-deficient board. For $n = 1$, such a board becomes an L-shape 3-cell piece called a tromino, as shown below.

(a) Use a divide-and-conquer technique to prove by induction that a quad-deficient board of size $2^n \times 2^n$, $n \geq 1$ can always be covered using some number of trominoes. (By covering we mean that every cell of the board must be covered by a tromino piece, and the pieces must not overlap.) Use diagram to help describing your algorithm and proof.

Solution: For the base, $n = 1$, the board is in the shape of a single tromino, thus it can be covered by a single tromino.

For $n \geq 2$, we will prove that if a $2^{n-1} \times 2^{n-1}$ board can be covered, then a $2^n \times 2^n$ board can also be covered. Consider a $2^n \times 2^n$ quad-deficient board. The board can be divided into four $2^{n-1} \times 2^{n-1}$ quad-deficient boards, as shown below. By the hypothesis, each of these smaller boards can be covered. Therefore the $2^n \times 2^n$ board can be covered.

(b) Let $f(n)$ be the number of tromino pieces used for covering a $2^n \times 2^n$ quad-deficient board. Write a recurrence for $f(n)$.

$$f(n) = \begin{cases} 1, & n = 1 \\ 4f(n-1), & n \geq 2 \end{cases}$$

The solution (though it was not asked for) is $f(n) = 4^{n-1}$. (You may verify this by either repeated substitution or by induction.) For example, $f(3) = 4^2 = 16$.

(c) Illustrate the covering produced by the algorithm for $n = 3$ (that is, $2^3 \times 2^3$ board).
5. (a) Insert the following sequence of elements into a Binary-Search-Tree (BST), starting with an empty tree: \((50, 90, 200, 25, 20, 10, 65, 35, 250)\).

(b) Delete element 80 in the following BST. (First complete the picture by carefully drawing a line from each node to its children, to get a valid BST.)

(c) What is the worst-case and average-case time complexity of BST operations SEARCH, INSERT, and DELETE?

Worst-Case Time = \(O(n)\)
Average-Case Time = \(O(\log n)\)