1. Given the following two functions:

    - $f(n) = 6n^3 + 4n^2 + 2$
    - $g(n) = 5n^2 + 9$

    Use L'Hopital's rule and limits to prove or disprove each of the following:

    - $f \in \Omega(g)$
    - $g \in \Theta(f)$

2. Rank the following functions from lowest asymptotic order to highest. List any two or more that are of the same order on the same line.

    - $2n^2 + 10n + 5$
    - $3n \log_2 n$
    - $4n + 10$
    - $3\sqrt{n}$
    - 2^n
    - $n^2 + 6n$
    - $2 \log_2 n$
    - $2n^3 + n^2 + 6$
    - 4^{n}
    - $\log_4 n$

3. Draw the recursion tree when $n = 12$, where n represents the length of the array, for the following recursive method:

    ```
    int sumSquares(int[] array, int first, int last) {
      if (first == last)
        return array[first] * array[first];
       int mid = (first + last) / 2;
       return sumSquares(array, first, mid) +
         sumSquares(array, mid + 1, last);
    }
    ```

    - Determine a formula that counts the numbers of nodes in the recursion tree.

    - What is the Big-Θ for execution time?

    - Determine a formula that expresses the height of the tree.

    - What is the Big-Θ for memory?

    - Write an iterative solution for this same problem and compare its efficiency with this recursive solution.

4. Using the recursive method in problem 3 and assuming n is the length of the array.

- Modify the recursion tree from the previous problem to show the amount of work on each activation and the row sums.

- Determine the initial conditions and recurrence equation.

- Determine the critical exponent.

- Apply the Little Master Theorem to solve that equation.

- Explain whether this algorithm optimal.