Assignment 1: Order Analysis, Basic Proof Techniques, and Programming Overview

Due: Sept 13, 4:30 PM

**Note:** Absolutely no late submissions will be accepted. 50% of the points for the order analysis and basic proof techniques and 50% of the points for the programming project.

**Order Analysis and Basic Proof Techniques**

1. Show that if \( f(n) = O(g(n)) \) and \( d(n) = O(h(n)) \), then \( f(n) + d(n) = O(g(n) + h(n)) \).
2. Show that \( O(\max\{f(n), g(n)\}) = O(f(n) + g(n)) \).
3. Show that if \( p(n) \) is a polynomial in \( n \), then \( \log p(n) = O(\log n) \).
4. Characterize the following summation (exactly) in terms of \( n \):
   \[
   \sum_{i=1}^{n} (3i + 4).
   \]
5. Show that \( \sum_{i=1}^{n} i^2 = O(n^3) \).
6. Show that \( \sum_{i=1}^{n} i/2^i < 2 \). (Hint: try to bound this sum term-by-term with a geometric progression.)
7. An \( n \)-degree polynomial \( p(x) \) is an equation of the form
   \[
   p(x) = \sum_{i=0}^{n} a_i x^i,
   \]
   where \( x \) is a real number and each \( a_i \) is a constant.
   
   (a) Describe a simple \( O(n^2) \) time method for computing \( p(x) \) for a particular value of \( x \).
   
   (b) Consider now a rewriting of \( p(x) \) as
   \[
   p(x) = a_0 + x(a_1 + x(a_2 + x(a_3 + \cdots + x(a_{n-1} + xa_n) \cdots))),
   \]
   which is known as Horner's method. Characterize, using the big-Oh notation, the number of multiplications and additions this method of evaluation uses.

**Programming Exercise**

Programming Project 8, Page 375 of your text.