1. (3 Points) The Mother of Series is very important, and you should recognize it in its many forms. For this problem, you must not directly use the formula for the geometric series.
   (a) Let \( S_4 = \frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \frac{1}{256} + \cdots = \sum_{i=1}^{\infty} (1/4)^i \). Use Figure 1(a) to determine the value of \( S_4 \).
   (b) Let \( S_3 = \frac{1}{3} + \frac{1}{9} + \frac{1}{27} + \frac{1}{81} + \cdots = \sum_{i=1}^{\infty} (1/3)^i \). Again, use Figure 1(b) to determine the value of \( S_3 \).
   (c) Generalize the arguments of (a) and (b) to \( S_k = \sum_{i=1}^{\infty} k^{-i} \).

2. (4 Points) Order the following 5 functions in order of increasing \( \Theta \)-order: (a) \( \log^2 n \), (b) \( n/\log^4 n \), (c) \( \sqrt{n} \), (d) \( n2^{-n} \), (e) \( \log \log n \).

3. (5 Points) Consider the expression \( E(x) := f(x)^{g(h(x))} \) where \( \{f, g, h\} = \{2^n, 1/n, \log n\} \). There are 6 = \( 3! \) possibilities for \( E(x) \). Determine the fastest and slowest functions among the six.

4. (4 Points) For each function, determine its growth type (this could mean “neither polynomial-type nor exponential-type”). You may use any known closure properties mentioned in the text, or argue from first principles:
   (a) \( 2^{n^2} \), (b) \( (\log \log n)^2 \) (c) \( n/\log n \),

5. (4 Points) Suppose
   \[ T_0(n) = 18T_0(n/6) + n^{1.5} \]
   and
   \[ T_1(n) = 32T_1(n/8) + n^{1.5} \]
   Which is the correct relation: \( T_0(n) = \Omega(T_1(n)) \) or \( T_0(n) = O(T_1(n)) \)? We want you to do this exercise without using a calculator or its equivalent; instead, use inequalities such as \( \log_8(x) < \log_6(x) \) (for \( x > 1 \)) and \( \log_6(2) < 1/2 \).

6. (2 Points) We want to improve on Karatsuba’s multiplication algorithm. We managed to subdivide a problem of size \( n \) into \( a \geq 2 \) subproblems of size \( n/4 \). After solving these \( a \) subproblems, we could combine their solutions in \( O(n) \) time to get the solution to the original problem of size \( n \). To beat Karatsuba, what is the maximum value \( a \) can have?
7. (4 Points) Recall our lecture on an algorithm to find the \( k \)-th largest of \( n \) elements (for any \( k = 1, \ldots, n \)). When \( k = \lceil n/2 \rceil \), we call this the \textbf{median problem}. Let \( M(n, k) \) be the number of comparisons for the \( k \)-th largest out of \( n \) elements. Also, let \( M(n) = \max \{ M(n, k) : k = 1, \ldots, n \} \).

(i) We showed in class that \( M(n) = M(n/5) + M(7n/10) + n \). Determine the watershed constant \( \alpha \) for this recurrence. We suggest you use a pocket calculator and determine \( \alpha \) up to 2 digits, using a simple binary search (one digit at a time).

(ii) Conclude from the Multiterm Master Theorem that \( M(n) = \Theta(n) \).

8. (4 Points) Jack has an algorithm whose complexity satisfies this recurrence:

\[
Ja(n) = 2Ja(n/3) + Ja(2n/5) + n
\]

while Jill’s algorithm satisfies

\[
Ji(n) = Ji(2n/3) + 2Ja(n/5) + n
\]

Use the Multiterm Master Theorem to decide who has the more efficient algorithm.

9. (0 Points) Use the Master Theorem \textit{(not the Multiterm Master Theorem)} to derive a sublinear upper bound on \( T(n) = 2T(n/3) + T(n/10) + 1 \). Recall some tricks in the text.