Problem 5 is worth 40% of this problem set, and problems 1-4 constitute the remaining 60%.

1 Problem 1

Indicate for each pair of expressions \((A, B)\) in the table below the relationship between \(A\) and \(B\). Your answer should be in the form of a table with a “yes” or “no” written in each box. For example, if \(A\) is \(O(B)\), then you should put a “yes” in the first box. If the base of a logarithm is not specified, you should assume it is base-2.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>(O)</th>
<th>(o)</th>
<th>(\Omega)</th>
<th>(\omega)</th>
<th>(\Theta)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\log_5 n)</td>
<td>(\log_4 n)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>((\log \log n)^2)</td>
<td>(\sqrt[3]{\log n})</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(2^{\log n})</td>
<td>(n^r)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(n!)</td>
<td>(n^n)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(\log(n!)</td>
<td>(\log(n^n))</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2 Problem 2

For all of the problems below, when asked to give an example, you should give a function mapping positive integers to positive integers.

- Show that if \(f\) is \(o(g)\), then \(f \cdot h\) is \(o(g \cdot h)\) for any positive function \(h\).
- Give a proof or a counterexample: if \(f\) is not \(O(g)\), then \(f\) is \(\Omega(g)\).
- Find (with proof) a function \(f\) such that \(f(2n)\) is \(O(f(n))\).
- Find (with proof) a function \(f\) such that \(f(n)\) is \(o(f(2n))\).
- Show that for all \(\epsilon > 0\), \(\log n\) is \(o(n^\epsilon)\).
3 Problem 3

InsertionSort is a simple sorting algorithm that works as follows on input \( A[0], \ldots, A[n-1] \):

\[
\text{InsertionSort}(A): \\
\text{for } i = 1 \text{ to } n-1 \\
\quad j = i \\
\quad \text{while } j > 0 \text{ and } A[j-1] > A[j] \\
\quad \quad \text{swap } A[j] \text{ and } A[j-1] \\
\quad j = j - 1
\]

Show that for any function \( T = T(n) \) satisfying \( T(n) = \Omega(n) \) and \( T(n) = O(n^2) \), there exists an infinite sequence of inputs \( \{A_n\}_{n=1}^{\infty} \) such that (1) \( A_n \) is an array of length \( n \), and (2) if \( f(n) \) denotes the running time of InsertionSort on \( A_n \), then \( f(n) = \Theta(T(n)) \).

4 Problem 4

Give asymptotic bounds for \( T(n) \) in each of the following recurrences. You may assume \( T(1) = 1 \) in each part below.

- \( T(n) = 7T(n/2) + n^3 \). You may assume \( n = 2^k \) for an integer \( k > 0 \).
- \( T(n) = 8T(n/2) + n^3 \). You may assume \( n = 2^k \) for an integer \( k > 0 \).
- \( T(n) = T(\sqrt[5]{n}) + \log n \). You may assume \( n = 2^{5^k} \) for an integer \( k \geq 0 \); also \( T(2) = 1 \).
- \( T(n) = 4T(n-2) + n \). You may assume \( n \) is odd.

5 Programming Problem