Quiz 2: Complexity, Sorting, Amortized Analysis, and Memoization

Copyright Oliver Serang, 2018
University of Montana Department of Computer Science

1. To which of the following sets does \( \frac{n^2}{\log(\log(n))} \) belong? Circle them.
   
   \[
   \begin{aligned}
   &O(n^2) \\
   &o(n^2) \\
   &\Omega(n^2) \\
   &\omega(n^2) \\
   &O\left(\frac{n^2}{\log\log(n)}\right) \\
   &o\left(\frac{n^2}{\log\log(n)}\right) \\
   &\Omega\left(\frac{n^2}{\log\log(n)}\right) \\
   &\omega\left(\frac{n^2}{\log\log(n)}\right) \\
   &O(n\log(n)) \\
   &o(n\log(n)) \\
   &\Omega(n\log(n)) \\
   &\omega(n\log(n)) \\
   &O(n) \\
   &o(n) \\
   &\Omega(n) \\
   &\omega(n)
   \end{aligned}
   \]

2. What is \( \lim_{n \to \infty} \log(\log(n)) \)?


<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Best-case</th>
<th>Worst-case</th>
<th>Average-case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection Sort</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Merge Sort</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quicksort</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
4. Circle the algorithm that best resembles merge sort:

```python
def f(array):
    f(array[:n/2])
    f(array[n/2:])
    do_some_work(array)
```

```python
def g(array):
    do_some_work(array)
    g(array[:n/2])
    g(array[n/2:])
```

5. “Tracing” is a garbage collection technique used by Java. In it, memory allocations are performed until the RAM is exhausted. Then, every block of memory no longer in use is deallocated. The worst-case runtime of garbage collection can be quite large, because it may need to deallocate many blocks. Using the accounting method, make an argument for why the amortized argument for garbage collection is not as bad. Assume that allocation and deallocation both run in $O(1)$.

6. Draw the call tree for a function $\text{pow}(a,b)$, which computes $a^b$ by computing $a^{b/2} \cdot a^{b/2}$. If we memoized this $\text{pow}(a,b)$ function to prevent redundant computations, what will be the runtime of calling $\text{pow}(a,b)$? You may assume $b$ is even in all recursive calls (i.e., $b$ is a power of 2).