This quiz has 3 questions, for a total of 30 points.

1. **9 points** Suppose that \( L \) is a Python “list” (array) of length \( n \) where \( n > 1 \). Categorize the worst-case execution time of each of the below operations as either

1. constant time (takes the same amount of time no matter what \( n \) is).
2. logarithmic time (takes time proportional to \( \log n \)).
3. linear time (takes time proportional to \( n \))
4. quadratic (takes time proportional to \( n^2 \))

Label each operation with the above item number.

- \( L.\text{insert}(n/2, 42) \)
- \( (2,1) \text{ in } L \)
- \( L[0] \)

**Solution:**

- (3), \( L.\text{insert}(n/2, 42) \) is linear time, (3 points)
- (3), \( (2,1) \text{ in } L \) is linear time, (3 points)
- (1), \( L[0] \) is constant time, (3 points)

2. **12 points** Complete the following implementation of the `delete` method in the following class that implements a doubly-linked list.

```python
class DLNode:
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None

class DoublyLinkedList:
    def __init__(self):
        self.head = None

    def delete(self, node):
        if node.prev:
            ___(a)___ = node.next
        else:
            self.head = ___(b)___
        if ___(c)___:
            node.next.prev = ___(d)___
```

**Solution:**

(a) \( \text{node.prev.next} \) (3 points)
(b) node.next (3 points)
(c) node.next (3 points)
(d) node.prev (3 points)

3. 9 points Let \( f(n) = \frac{n}{2} - 1 \) and \( g(n) = n \). Give the definition of \( \Omega \) and prove that \( f(n) \in \Omega(g(n)) \). (Use the other side of this paper for the proof.)

Solution:

\[
\Omega(g(n)) = \{ f(n) \mid \exists n_0. \forall n \geq n_0. \exists c. 0 \leq c g(n) \leq f(n) \}
\]

We need to choose a \( c \) such that \( cn \) becomes less than \( \frac{n}{2} - 1 \) at some point. We choose \( c = \frac{1}{4} \). Next we need to find out at what point \( \frac{1}{4}n \) is equal to or less than \( \frac{n}{2} - 1 \), so we chart those out:

<table>
<thead>
<tr>
<th>( n )</th>
<th>( \frac{n}{2} - 1 )</th>
<th>( \frac{1}{4}n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.5</td>
<td>( \frac{1}{4} )</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>( \frac{1}{2} )</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>( \frac{3}{4} )</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1.5</td>
<td>( \frac{5}{4} )</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>( \frac{3}{2} )</td>
</tr>
</tbody>
</table>

So it looks like \( n_0 = 4 \) is a good choice. We are now ready to give the proof.

To show that \( \frac{n}{2} - 1 \in \Omega(n) \), we need to show that

\[
\exists n_0. \forall n \geq n_0. \exists c. 0 \leq c n \leq \frac{n}{2} - 1
\]

We choose \( n_0 = 4 \) and \( c = \frac{1}{4} \). So we need to prove that

\[
\forall n \geq 4. 0 \leq \frac{n}{4} \leq \frac{n}{2} - 1
\]

We multiple both sides by 4 to get the following

\[
\forall n \geq 4. 0 \leq n \leq 2n - 4
\]

We can prove this by induction. For the base case, we have \( n = 4 \). So \( 0 \leq 4 \leq 4 \).

For the induction step, we may assume that \( 0 \leq k \leq 2k - 4 \) for some \( k \geq 4 \). We add 1 to both sides to get

\[
k + 1 \leq 2k - 3
\]

We need to show that \( k + 1 \leq 2(k + 1) - 4 \). But \( 2(k + 1) - 4 = 2k - 2 \), and \( 2k - 3 \leq 2k - 2 \), so the proof is complete.