1 Setup

This document will give an inductive proof that the below implementation of `factorial` has a runtime in $O(n)$.

Code:

```
public int factorial(int n) {
    if (n <= 1) {
        return 1;
    } else {
        return n * factorial(n-1);
    }
}
```

Growth Function:

$$T(n) = \begin{cases} c_0 & \text{if } n \leq 1 \\ T(n-1) + c_1 & \text{otherwise} \end{cases}$$

Where c_0 and c_1 are constants.

Hypothesis:

$$T(n) \in O(n)$$

or equivalently, we must show that $P(n)$ is true for all $n \geq n_0$, where $n_0 \geq 0$ and $P(n)$ is defined as:

$$P(n) : \text{There exists a constant}, c > 0, \text{ such that } T(n) \leq c \cdot n$$

2 Base Case Proof

To prove the base case, we need to prove our hypthesis holds for a fixed value of n. Specifically, we will show that $P(1)$ is true.

$$T(1) \stackrel{?}{\leq} c \cdot 1 \tag{1}$$

$$c_0 \stackrel{?}{\leq} c \tag{2}$$

Therefore, as long as we pick $c \geq c_0$ then the inequality holds, therefore P(1) is true.

3 Inductive Proof

We will now show that $P(n-1) \implies P(n)$.
To do so, we will start by assuming that $P(n-1)$ is true. More specifically we will assume that:

$$\text{There exists a constant}, c > 0, \text{ such that } T(n-1) \leq c \cdot (n-1)$$

We must now show that under this assumption, $P(n)$ must also be true.

$$T(n) \stackrel{?}{\leq} c \cdot n \tag{3}$$

$$T(n-1) + c_1 \stackrel{?}{\leq} c \cdot n \tag{4}$$

$$T(n-1) + c_1 \leq c \cdot (n-1) + c_1 \stackrel{?}{\leq} c \cdot n \tag{5}$$

$$c \cdot n - c + c_1 \stackrel{?}{\leq} c \cdot n \tag{6}$$

$$c_1 \stackrel{?}{\leq} c \tag{7}$$

Note: Line (4) was obtained by substituting the definition of $T(n)$
Note: Line (5) relies on our assumption and the principle of transitivity
Therefore, as long as we pick $c \geq c_1$, $P(n-1) \implies P(n)$.

4 Conclusion

Our goal was to show that $P(n)$ is true for all values of n greater than some non-negative constant. In the base case, we got a constraint on c that stated $c \geq c_0$. In the inductive step we got an additional constraint that $c \geq c_1$. We can satify both of these constraints by picking a value for c that is greater than both c_0 and c_1, ie: $c = c_0 + c_1$.

We then have shown that if $c = c_0 + c_1$, $P(n)$ is true when $n = 1$ (in the Base Case). Furthermore, the inductive step proves that if $c = c_0 + c_1$, then $P(n-1) \implies P(n)$. Therefore $P(n)$ is true for all $n \geq 1$, Therefore $T(n) \in O(n)$.