Big-Oh Notation

Let $f$ and $g$ be functions from positive numbers to positive numbers. $f(n)$ is $O(g(n))$ if there are positive constants $C$ and $k$ such that:

$$f(n) \leq C g(n) \text{ whenever } n > k$$

$$f(n) \text{ is } O(g(n)) \equiv \exists C \exists k \forall n \ (n > k \rightarrow f(n) \leq C g(n))$$

To prove big-Oh, choose values for $C$ and $k$ and prove $n > k$ implies $f(n) \leq C g(n)$.

________________________________________________________

Standard Method to Prove Big-Oh

1. Choose $k = 1$.
2. Assuming $n > 1$, find/derive a $C$ such that

$$\frac{f(n)}{g(n)} \leq \frac{C g(n)}{g(n)} = C$$

This shows that $n > 1$ implies $f(n) \leq C g(n)$.

Keep in mind:

- $n > 1$ implies $1 < n$, $n < n^2$, $n^2 < n^3$, …
- “Increase” numerator to “simplify” fraction.
Proving Big-Oh: Example 1

Show that $f(n) = n^2 + 2n + 1$ is $O(n^2)$.

Choose $k = 1$.

Assuming $n > 1$, then

$$\frac{f(n)}{g(n)} = \frac{n^2 + 2n + 1}{n^2} < \frac{n^2 + 2n^2 + n^2}{n^2} = 4$$

Choose $C = 4$. Note that $2n < 2n^2$ and $1 < n^2$.

Thus, $n^2 + 2n + 1$ is $O(n^2)$ because $n^2 + 2n + 1 \leq 4n^2$ whenever $n > 1$.

Proving Big-Oh: Example 2

Show that $f(n) = 3n + 7$ is $O(n)$.

Choose $k = 1$.

Assuming $n > 1$, then

$$\frac{f(n)}{g(n)} = \frac{3n + 7}{n} < \frac{3n + 7n}{n} = \frac{10n}{n} = 10$$

Choose $C = 10$. Note that $7 < 7n$.

Thus, $3n + 7$ is $O(n)$ because $3n + 7 \leq 10n$ whenever $n > 1$. 
Proving Big-Oh: Example 3

Show that \( f(n) = (n + 1)^3 \) is \( O(n^3) \).

Choose \( k = 1 \).

Assuming \( n > 1 \), then
\[
\frac{f(n)}{g(n)} = \frac{(n + 1)^3}{n^3} < \frac{(n + n)^3}{n^3} = \frac{8n^3}{n^3} = 8
\]

Choose \( C = 8 \). Note that \( n + 1 < n + n \) and \((n+n)^3 = (2n)^3 = 8n^3 \). Thus, \((n+1)^3 \) is \( O(n^3) \) because \((n + 1)^3 \leq 8n^3 \) whenever \( n > 1 \).

Proving Big-Oh: Example 4

Show that \( f(n) = \sum_{i=1}^{n} i \) is \( O(n^2) \).

Choose \( k = 1 \).

Assuming \( n > 1 \), then
\[
\frac{f(n)}{g(n)} = \frac{\sum_{i=1}^{n} i}{n^2} \leq \frac{\sum_{i=1}^{n} n}{n^2} = \frac{n^2}{n^2} = 1
\]

Choose \( C = 1 \). Note that \( i \leq n \) because \( n \) is the upper limit. Thus, \( \sum_{i=1}^{n} i \) is \( O(n^2) \) because \( \sum_{i=1}^{n} i \leq n^2 \) whenever \( n > 1 \).
How to Show Not Big-Oh

\[ f(n) \text{ is not } O(g(n)) \equiv \]
\[ \forall C \forall k \exists n \,(n > k \land f(n) > C g(n)) \]

Need to prove for all values of \( C \) and \( k \).

\( C \) and \( k \) cannot be replaced with constants.

Choose \( n \) based on \( C \) and \( k \).

Prove that this choice implies
\[ n > k \land f(n) > C g(n) \]

Standard Method to Prove Not-Big-Oh:
1. Assume \( n > 1 \).

2. Show:
\[ \frac{f(n)}{g(n)} \geq \frac{h(n) g(n)}{g(n)} = h(n) \]
where \( h(n) \) is strictly increasing to \( \infty \).

3. \( n > h^{-1}(C') \) implies \( h(n) > C \), which implies \( f(n) > C g(n) \).

So choosing \( n > 1 \), \( n > k \), and \( n > h^{-1}(C') \) implies \( n > k \land f(n) > C g(n) \).
Proving Not Big-Oh: Example 1

Show that \( f(n) = n^2 - 2n + 1 \) is not \( O(n) \).

Assume \( n > 1 \), then

\[
\frac{f(n)}{g(n)} = \frac{n^2 - 2n + 1}{n} > \frac{n^2 - 2n}{n} = n - 2
\]

\( n > C + 2 \) implies \( n - 2 > C \) and \( f(n) > Cn \).

So choosing \( n > 1 \), \( n > k \), and \( n > C + 2 \) implies \( n > k \land f(n) > Cn \).

• “Decrease” numerator to “simplify” fraction.

Proving Not Big-Oh: Example 2

Show that \( f(n) = (n - 1)^3 \) is not \( O(n^2) \).

Assume \( n > 1 \), then:

\[
\frac{f(n)}{g(n)} = \frac{n^3 - 3n^2 + 3n - 1}{n^2} > \frac{n^3 - 3n^2 - 1}{n^2}
\]

\[
> \frac{n^3 - 3n^2 - n^2}{n^2} = n - 4
\]

\( n > C + 4 \) implies \( n - 4 > C \) and \( f(n) > Cn^2 \).

Choosing \( n > 1 \), \( n > k \), and \( n > C + 4 \) implies \( n > k \land f(n) > Cn^2 \).
Proving Not Big-Oh: Example 3

Show that \( f(n) = \lfloor n^2/2 \rfloor \) is not \( O(n) \).

Assume \( n > 1 \), then:

\[
\frac{f(n)}{g(n)} = \frac{\lfloor n^2/2 \rfloor}{n} > \frac{n^2/2 - 1}{n} > \frac{n^2/2 - n}{n} = n/2 - 1
\]

\( n > 2C + 2 \rightarrow n/2 - 1 > C \) and \( f(n) > Cn \).

Choosing \( n > 1 \), \( n > k \), and \( n > 2C + 2 \) implies \( n > k \land f(n) > Cn \).