Asymptotic Analysis
Algorithm A: \( f.n = 25n + 93 \)

Algorithm B: \( f.n = 2n^2 + 4n - 3 \)

Which is better?
Algorithm A: $f.n = 25n + 93$

Algorithm B: $f.n = 2n^2 + 4n - 3$
Definition of asymptotic upper bound: For a given function $g.n$, $O(g.n)$, pronounced “big-oh of $g$ of $n$”, is the set of functions

$$\{ f.n \mid (\exists c, n_0 \mid c > 0 \land n_0 > 0 : (\forall n \mid n \geq n_0 : 0 \leq f.n \leq c \cdot g.n) ) \}.$$ 

O-notation: $f.n = O(g.n)$ means function $f.n$ is in the set $O(g.n)$. 
Design Patterns for Data Structures

\[ 25n + 93 = O(n) \]
Design Patterns for Data Structures

\[ 25n + 93 = O(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that \( 25n + 93 \leq cn \) for \( n \geq n_0 \).
Design Patterns for Data Structures

\[ 25n + 93 = O(n) \]

*Proof:* Must prove that there exist positive constants \( c, n_0 \) such that
\[ 25n + 93 \leq cn \quad \text{for } n \geq n_0. \]

\[ 25n + 93 \]
Design Patterns for Data Structures

\[ 25n + 93 = O(n) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 25n + 93 \leq cn \quad \text{for } n \geq n_0. \]

\[ 25n + 93 \leq \langle \text{Replacing 93 with a larger value, provided } n \geq 93 \rangle \]
\[ 25n + 93 = O(n) \]

\textit{Proof}: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 25n + 93 \leq cn \quad \text{for } n \geq n_0. \]
\[
25n + 93 \\
\leq \langle \text{Replacing } 93 \text{ with a larger value, provided } n \geq 93 \rangle \\
25n + n
Design Patterns for Data Structures

\[ 25n + 93 = O(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that 
\[ 25n + 93 \leq cn \quad \text{for } n \geq n_0. \]

\[ 25n + 93 \]
\[ \leq \langle \text{Replacing 93 with a larger value, provided } n \geq 93 \rangle \]
\[ 25n + n \]
\[ = \langle \text{Math} \rangle \]
\[ 25n + 93 = O(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 25n + 93 \leq cn \quad \text{for} \quad n \geq n_0. \]

\[
\begin{align*}
25n + 93 & \leq (\text{Replacing 93 with a larger value, provided } n \geq 93) \\
25n + n & = (\text{Math}) \\
26n &
\end{align*}
\]
\[ 25n + 93 = O(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 25n + 93 \leq cn \quad \text{for} \quad n \geq n_0. \]

\[ 25n + 93 \]
\[ \leq \langle \text{Replacing 93 with a larger value, provided } n \geq 93 \rangle \]
\[ 25n + n \]
\[ = \langle \text{Math} \rangle \]
\[ 26n \]
\[ = \langle \text{Provided } c = 26 \rangle \]
$$25n + 93 = O(n)$$

**Proof**: Must prove that there exist positive constants $c, n_0$ such that

$$25n + 93 \leq cn \quad \text{for } n \geq n_0.$$ 

$$25n + 93 \leq \langle \text{Replacing 93 with a larger value, provided } n \geq 93 \rangle$$

$$25n + n \leq \langle \text{Math} \rangle$$

$$26n \leq \langle \text{Provided } c = 26 \rangle$$

$$cn$$
\[ 25n + 93 = O(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 25n + 93 \leq cn \quad \text{for} \quad n \geq n_0. \]

\[ 25n + 93 \leq \langle \text{Replacing 93 with a larger value, provided} \ n \geq 93 \rangle \]

\[ 25n + n = \langle \text{Math} \rangle \]

\[ 26n = \langle \text{Provided} \ c = 26 \rangle \]

\[ cn \]

So, \( c = 26, \ n_0 = 93. \]
\[2n^2 + 4n - 3 = O(n^2)\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = O(n^2) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that \( 2n^2 + 4n - 3 \leq cn^2 \) for \( n \geq n_0 \).
\[ 2n^2 + 4n - 3 = O(n^2) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \leq cn^2 \quad \text{for} \quad n \geq n_0. \]
\[ 2n^2 + 4n - 3 \]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = O(n^2) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0. \]

\[ 2n^2 + 4n - 3 \leq \langle \text{Eliminating a negative value} \rangle \]
\[ 2n^2 + 4n - 3 = O(n^2) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
2n^2 + 4n - 3 &\leq \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n &\leq \langle \text{Eliminating a negative value} \rangle 
\end{align*}
\]
\[ 2n^2 + 4n - 3 = O(n^2) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
2n^2 + 4n - 3 & \\
\leq & \quad \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n & \\
\leq & \quad \langle \text{Replacing } 4n \text{ with a larger value, provided } n \geq 1 \rangle
\end{align*}
\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = O(n^2) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
2n^2 + 4n - 3 & \\
\leq \quad \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n & \\
\leq \quad \langle \text{Replacing } 4n \text{ with a larger value, provided } n \geq 1 \rangle \\
2n^2 + 4n \cdot n &
\end{align*}
\]
Design Patterns for Data Structures

\[2n^2 + 4n - 3 = O(n^2)\]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that 
\[2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0.\]

\[
2n^2 + 4n - 3 \\
\leq \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n \\
\leq \langle \text{Replacing } 4n \text{ with a larger value, provided } n \geq 1 \rangle \\
2n^2 + 4n \cdot n \\
= \langle \text{Math} \rangle
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = O(n^2) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
2n^2 + 4n - 3 & \leq \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n & \leq \langle \text{Replacing } 4n \text{ with a larger value, provided } n \geq 1 \rangle \\
2n^2 + 4n \cdot n & = \langle \text{Math} \rangle \\
6n^2 &
\end{align*}
\]
Design Patterns for Data Structures

\[2n^2 + 4n - 3 = O(n^2)\]

**Proof**: Must prove that there exist positive constants \(c, n_0\) such that
\[2n^2 + 4n - 3 \leq cn^2 \quad \text{for} \quad n \geq n_0.\]

\[
\begin{align*}
2n^2 + 4n - 3 & \\
\leq & \quad \langle\text{Eliminating a negative value}\rangle \\
2n^2 + 4n & \\
\leq & \quad \langle\text{Replacing } 4n \text{ with a larger value, provided } n \geq 1\rangle \\
2n^2 + 4n \cdot n & \\
= & \quad \langle\text{Math}\rangle \\
6n^2 & \\
= & \quad \langle\text{Provided } c = 6\rangle
\end{align*}
\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = O(n^2) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that \( 2n^2 + 4n - 3 \leq cn^2 \) for \( n \geq n_0 \).

\[
2n^2 + 4n - 3 \\
\leq \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n \\
\leq \langle \text{Replacing } 4n \text{ with a larger value, provided } n \geq 1 \rangle \\
2n^2 + 4n \cdot n \\
= \langle \text{Math} \rangle \\
6n^2 \\
= \langle \text{Provided } c = 6 \rangle \\
cn^2 \]
\[2n^2 + 4n - 3 = O(n^2)\]

**Proof**: Must prove that there exist positive constants \(c, n_0\) such that 
\[2n^2 + 4n - 3 \leq cn^2 \quad \text{for } n \geq n_0.\]

\[
\begin{align*}
2n^2 + 4n - 3 & \leq \langle \text{Eliminating a negative value} \rangle \\
2n^2 + 4n & \leq \langle \text{Replacing } 4n \text{ with a larger value, provided } n \geq 1 \rangle \\
2n^2 + 4n \cdot n & = \langle \text{Math} \rangle \\
6n^2 & = \langle \text{Provided } c = 6 \rangle \\
& \quad cn^2
\end{align*}
\]

So, \(c = 6, n_0 = 1\). ■
Definition of asymptotic lower bound: For a given function $g.n$, $\Omega(g.n)$, pronounced “big-omega of $g$ of $n$”, is the set of functions

\[
\{ f.n \mid (\exists c,n_0 \mid c > 0 \land n_0 > 0 : (\forall n \mid n \geq n_0 : 0 \leq c \cdot g.n \leq f.n) ) \}.
\]

$\Omega$-notation: $f.n = \Omega(g.n)$ means function $f.n$ is in the set $\Omega(g.n)$. 
Design Patterns for Data Structures

\[ 25n + 93 = \Omega(n) \]
\[ 25n + 93 = \Omega(n) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that \( 25n + 93 \geq cn \) for \( n \geq n_0 \).
Design Patterns for Data Structures

\[ 25n + 93 = \Omega(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that \( 25n + 93 \geq cn \) for \( n \geq n_0 \).

\[ 25n + 93 \]
\[ 25n + 93 = \Omega(n) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 25n + 93 \geq cn \quad \text{for } n \geq n_0. \]

\[ 25n + 93 \]
\[ \geq \langle \text{Eliminating a positive value} \rangle \]
25n + 93 = \Omega (n)

Proof: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 25n + 93 \geq cn \quad \text{for } n \geq n_0. \]

\[ 25n + 93 \geq (\text{Eliminating a positive value}) \]
\[ 25n \]
\[25n + 93 = \Omega(n)\]

**Proof**: Must prove that there exist positive constants \(c, n_0\) such that \(25n + 93 \geq cn\) for \(n \geq n_0\).

\[
\begin{align*}
25n + 93 &\geq (\text{Eliminating a positive value}) \\
25n &\geq (\text{Provided } c = 25)
\end{align*}
\]
\[ 25n + 93 = \Omega(n) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that \( 25n + 93 \geq cn \) for \( n \geq n_0 \).

\[
\begin{align*}
25n + 93 \\
\geq & \quad \langle \text{Eliminating a positive value} \rangle \\
25n \\
= & \quad \langle \text{Provided } c = 25 \rangle \\
\quad cn
\end{align*}
\]
\[25n + 93 = \Omega(n)\]

**Proof**: Must prove that there exist positive constants \(c, n_0\) such that
\[25n + 93 \geq cn \quad \text{for } n \geq n_0.\]

\[
\begin{align*}
25n + 93 & \geq \langle \text{Eliminating a positive value} \rangle \\
25n & \geq \langle \text{Provided } c = 25 \rangle \\
\end{align*}
\]

So, \(c = 25\), and any positive \(n_0\) is possible, say \(n_0 = 1.\)  □
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = \Omega (n^2) \]
\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

*Proof*: Must prove that there exist positive constants \(c, n_0\) such that \(2n^2 + 4n - 3 \geq cn^2\) for \(n \geq n_0\).
\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \geq cn^2 \quad \text{for } n \geq n_0. \]

\[ 2n^2 + 4n - 3 \]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \geq cn^2 \]
for \( n \geq n_0 \).

\[
2n^2 + 4n - 3 \\
\geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle
\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \geq cn^2 \quad \text{for } n \geq n_0. \]
\[
2n^2 + 4n - 3 \geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n \]
\[2n^2 + 4n - 3 = \Omega(n^2)\]

**Proof**: Must prove that there exist positive constants \(c, n_0\) such that
\[2n^2 + 4n - 3 \geq cn^2\] for \(n \geq n_0\).

\[
2n^2 + 4n - 3 \\
\geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n \\
= \langle \text{Math} \rangle
\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 2n^2 + 4n - 3 \geq cn^2 \quad \text{for } n \geq n_0. \]

\[
2n^2 + 4n - 3 \\
\geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n \\
= \langle \text{Math} \rangle \\
2n^2 + n
\]
\[2n^2 + 4n - 3 = \Omega(n^2)\]

**Proof**: Must prove that there exist positive constants \(c, n_0\) such that
\[2n^2 + 4n - 3 \geq cn^2 \quad \text{for } n \geq n_0.\]

\[
2n^2 + 4n - 3 \\
\geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n \\
= \langle \text{Math} \rangle \\
2n^2 + n \\
\geq \langle \text{Eliminating a positive value, provided } n \geq 0 \rangle
\]
\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

**Proof:** Must prove that there exist positive constants \( c, n_0 \) such that \( 2n^2 + 4n - 3 \geq cn^2 \) for \( n \geq n_0 \).

\[
\begin{align*}
2n^2 + 4n - 3 & \\
\geq & \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n & \\
= & \langle \text{Math} \rangle \\
2n^2 + n & \\
\geq & \langle \text{Eliminating a positive value, provided } n \geq 0 \rangle \\
2n^2 & 
\end{align*}
\]
$2n^2 + 4n - 3 = \Omega(n^2)$

**Proof:** Must prove that there exist positive constants $c, n_0$ such that $2n^2 + 4n - 3 \geq cn^2$ for $n \geq n_0$.

\[
\begin{align*}
2n^2 + 4n - 3 & \geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n & = \langle \text{Math} \rangle \\
2n^2 + n & \geq \langle \text{Eliminating a positive value, provided } n \geq 0 \rangle \\
2n^2 & = \langle \text{Provided } c = 2 \rangle
\end{align*}
\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

*Proof:* Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \geq cn^2 \quad \text{for} \quad n \geq n_0. \]

\[
\begin{align*}
2n^2 + 4n - 3 & \geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n & = \langle \text{Math} \rangle \\
2n^2 + n & \geq \langle \text{Eliminating a positive value, provided } n \geq 0 \rangle \\
2n^2 & = \langle \text{Provided } c = 2 \rangle \\
cn^2 &
\end{align*}
\]
Design Patterns for Data Structures

\[ 2n^2 + 4n - 3 = \Omega(n^2) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 2n^2 + 4n - 3 \geq cn^2 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
2n^2 + 4n - 3 & \geq \langle \text{Increasing a negative value, provided } n \geq 1 \rangle \\
2n^2 + 4n - 3n & \geq \langle \text{Math} \rangle \\
2n^2 + n & \geq \langle \text{Eliminating a positive value, provided } n \geq 0 \rangle \\
2n^2 & \geq \langle \text{Provided } c = 2 \rangle \\
cn^2 & \end{align*}
\]

So, \( c = 2, n_0 = 1. \)
\[ 6n^3 - 7n = \Omega(n^3) \]
\[ 6n^3 - 7n = \Omega(n^3) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]
\[ 6n^3 - 7n = \Omega(n^3) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[ 6n^3 - 7n \]
\[ 6n^3 - 7n = \Omega(n^3) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
6n^3 - 7n & \geq ( \quad ) \\
5n^3 & \end{align*}
\]
\[ 6n^3 - 7n = \Omega(n^3) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that
\[
6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0.
\]

\[
\begin{align*}
6n^3 - 7n &\geq (6n^3 - 7n \geq 5n^3, \\
5n^3 &
\end{align*}
\]
\[ 6n^3 - 7n = \Omega(n^3) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
6n^3 - 7n &
\geq \langle 6n^3 - 7n \geq 5n^3, \ n^3 \geq 7n, \\
5n^3 &
\end{align*}
\]
$$6n^3 - 7n = \Omega(n^3)$$

Proof: Must prove that there exist positive constants $c, n_0$ such that $6n^3 - 7n \geq cn^3$ for $n \geq n_0$.

$$6n^3 - 7n \geq \left\langle 6n^3 - 7n \geq 5n^3, \ n^3 \geq 7n, \ n^2 \geq 7, \ 5n^3 \right\rangle$$
\[ 6n^3 - 7n = \Omega(n^3) \]

*Proof:* Must prove that there exist positive constants \( c, n_0 \) such that
\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]

\begin{align*}
6n^3 - 7n & \geq 6n^3 - 7n \geq 5n^3, \\
& \geq 6n^3 - 7n \geq 5n^3, \ n^3 \geq 7n, \ n^2 \geq 7, \ n \geq \sqrt{7} \tag{5n^3} \\
\end{align*}
\[ 6n^3 - 7n = \Omega(n^3) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that

\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
6n^3 - 7n &\geq \langle 6n^3 - 7n \geq 5n^3, \ n^3 \geq 7n, \ n^2 \geq 7, \ n \geq \sqrt{7} \rangle \\
5n^3 &\geq \langle \text{Provided } c = 5 \rangle
\end{align*}
\]
\[ 6n^3 - 7n = \Omega(n^3) \]

**Proof:** Must prove that there exist positive constants \( c, n_0 \) such that

\[ 6n^3 - 7n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
6n^3 - 7n \\
\geq \langle 6n^3 - 7n \geq 5n^3, \ n^3 \geq 7n, \ n^2 \geq 7, \ n \geq \sqrt{7} \rangle \\
5n^3 \\
= \langle \text{Provided } c = 5 \rangle \\
\end{align*}
\]
\[6n^3 - 7n = \Omega(n^3)\]

*Proof*: Must prove that there exist positive constants \(c, n_0\) such that \(6n^3 - 7n \geq cn^3\) for \(n \geq n_0\).

\[
\begin{align*}
6n^3 - 7n &\geq \langle 6n^3 - 7n \geq 5n^3, n^3 \geq 7n, n^2 \geq 7, n \geq \sqrt{7} \rangle \\
5n^3 &\geq \langle \text{Provided } c = 5 \rangle \\
cn^3 &\text{ So, } c = 5, n_0 = \sqrt{7}. \quad \square
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]
\[ n^3 - 5n = \Omega(n^3) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]
\[ n^3 - 5n = \Omega(n^3) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]
Design Patterns for Data Structures

\[ n^3 - 5n = \Omega (n^3) \]

**Proof**: Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
 n^3 - 5n \\
\geq \langle \quad \rangle \\
0.5n^3
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that 
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
& n^3 - 5n \\
& \geq (n^3 - 5n \geq 0.5n^3, \\
& 0.5n^3)
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
n^3 - 5n & \\
\geq & \quad \langle n^3 - 5n \geq 0.5n^3, \ 0.5n^3 \geq 5n, \rangle \\
0.5n^3 & 
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]

Proof: Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
  n^3 - 5n &\geq \langle n^3 - 5n \geq 0.5n^3, \ 0.5n^3 \geq 5n, \ n^2 \geq 10, \rangle \\
  0.5n^3 &\geq \n
\]
\[ n^3 - 5n = \Omega(n^3) \]

*Proof*: Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
n^3 - 5n &\geq n^3 - 5n \geq 0.5n^3, \quad 0.5n^3 \geq 5n, \quad n^2 \geq 10, \quad n \geq \sqrt{10} \\
&\geq 0.5n^3
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]

**Proof:** Must prove that there exist positive constants \( c, n_0 \) such that
\[ n^3 - 5n \geq cn^3 \quad \text{for } n \geq n_0. \]

\[
\begin{align*}
& n^3 - 5n \\
& \geq \langle n^3 - 5n \geq 0.5n^3, \ 0.5n^3 \geq 5n, \ n^2 \geq 10, \ n \geq \sqrt{10} \rangle \\
& 0.5n^3 \\
& = \langle \text{Provided } c = 0.5 \rangle
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]

**Proof:** Must prove that there exist positive constants \( c, n_0 \) such that \( n^3 - 5n \geq cn^3 \) for \( n \geq n_0 \).

\[
\begin{align*}
n^3 - 5n &
\geq \langle n^3 - 5n \geq 0.5n^3, \ 0.5n^3 \geq 5n, \ n^2 \geq 10, \ n \geq \sqrt{10} \rangle \\
0.5n^3 &
= \langle \text{Provided } c = 0.5 \rangle \\
cn^3 &
\end{align*}
\]
\[ n^3 - 5n = \Omega(n^3) \]

\textit{Proof}: Must prove that there exist positive constants \( c, n_0 \) such that \( n^3 - 5n \geq cn^3 \) for \( n \geq n_0 \).

\[
\begin{align*}
    & n^3 - 5n \\
    \geq & \quad \langle n^3 - 5n \geq 0.5n^3, \ 0.5n^3 \geq 5n, \ n^2 \geq 10, \ n \geq \sqrt{10} \rangle \\
    & 0.5n^3 \\
    = & \quad \langle \text{Provided } c = 0.5 \rangle \\
    & cn^3 \\
\end{align*}
\]

So, \( c = 0.5, \ n_0 = \sqrt{10} \).
Definition of asymptotic tight bound: For a given function $g.n$, $\Theta(g.n)$, pronounced “big-theta of $g$ of $n$”, is the set of functions

$$\{f.n \mid (\exists c_1, c_2, n_0 \mid c_1 > 0 \land c_2 > 0 \land n_0 > 0 : \\
(\forall n \mid n \geq n_0 : 0 \leq c_1 \cdot g.n \leq f.n \leq c_2 \cdot g.n) \}.$$ 

$\Theta$-notation: $f.n = \Theta(g.n)$ means function $f.n$ is in the set $\Theta(g.n)$. 
**Definition of asymptotic tight bound:** For a given function $g.n$, $\Theta(g.n)$, pronounced “big-theta of $g$ of $n$”, is the set of functions

$$\{f.n \mid (\exists c_1, c_2, n_0 \mid c_1 > 0 \land c_2 > 0 \land n_0 > 0 : \\
(\forall n \mid n \geq n_0 : 0 \leq c_1 \cdot g.n \leq f.n \leq c_2 \cdot g.n ))\}.$$ 

**$\Theta$-notation:** $f.n = \Theta(g.n)$ means function $f.n$ is in the set $\Theta(g.n)$.

**Theorem:** $f.n = \Theta(g.n)$ if and only if $f.n = O(g.n)$ and $f.n = \Omega(g.n)$. 
Definition of asymptotic tight bound: For a given function \( g.n \), \( \Theta(g.n) \), pronounced “big-theta of \( g \) of \( n \)”, is the set of functions

\[
\{ f.n \mid (\exists c_1, c_2, n_0 \mid c_1 > 0 \land c_2 > 0 \land n_0 > 0 : \\
(\forall n \mid n \geq n_0 : 0 \leq c_1 \cdot g.n \leq f.n \leq c_2 \cdot g.n) \}.
\]

\( \Theta \)-notation: \( f.n = \Theta(g.n) \) means function \( f.n \) is in the set \( \Theta(g.n) \).

Theorem: \( f.n = \Theta(g.n) \) if and only if \( f.n = O(g.n) \) and \( f.n = \Omega(g.n) \).

\[
25n + 93 = \Theta(n)
\]

\[
2n^2 + 4n - 3 = \Theta(n^2)
\]
Interpretation of big theta

\[ f(n) = 2n^2 + 4n - 3 \]

\[ f(n) = 6n^2 \]

\[ f(n) = 2n^2 \]
Polynomial bound: Define an asymptotically positive polynomial $p.n$ of degree $d$ to be

$$p.n = (\sum i | 0 \leq i \leq d : a_i n^i)$$

where the constants $a_0, a_1, \ldots, a_d$ are the coefficients of the polynomial and $a_d > 0$. Then $p.n = \Theta(n^d)$. 