Big O Rules (Facts):

1. **Transitivity of Big O (i.e., the Chain Rule):** $f(n) \in \mathbf{O}(g(n))$ and $g(n) \in \mathbf{O}(h(n)) \Rightarrow f(n) \in \mathbf{O}(h(n))$.
2. **Strange But True Log Domination Rule:** $(\log n)^r \in \mathbf{O}(n^s)$ for all constants r, and all constants $s > 0$.
3. **Polynomial Rule:** $p(n) \in \mathbf{O}(q(n))$, whenever $p(n)$ is a polynomial in n, of degree k, and $q(n)$ is a polynomial in n, of degree t, where k and t are constants and $k \leq t$.
4. **Product Rule:** $f_1(n) \in \mathbf{O}(g_1(n))$ and $f_2(n) \in \mathbf{O}(g_2(n)) \Rightarrow f_1(n) \cdot f_2(n) \in \mathbf{O}(g_1(n) \cdot g_2(n))$.
5. **Removal of Constant Factors Rule :** $c \cdot f(n) \in \mathbf{O}(f(n))$ for all constants $c > 0$.
6. **Log Base is Irrelevant Rule:** $\log_a n \in \mathbf{O}(\log_b n)$ for all constants $a, b > 0$ and $a, b > 1$.
7. **Reciprocal Rule:** $f(n) \in \mathbf{O}((g)n)$, then $\frac{1}{g(n)} \in \mathbf{O}(\frac{1}{f(n)})$.
8. **Sum Rule:** If $f_1(n) \in \mathbf{O}(g(n))$ and $f_2(n) \in \mathbf{O}(g(n))$ then $f_1(n) + f_2(n) \in \mathbf{O}(g(n))$.
9. **Negative Terms rule:** if $f(n) \in \mathbf{O}(g(n))$, and $h(n)$ is a function that is always positive for all $n \geq 0$, then $f(n) - h(n) \in \mathbf{O}(g(n))$. You must prove that $h(n)$ is always positive for all $n \geq 0$.
10. **Proved separately:** You can bring into any proof "by the Rules" any specific statement that you have proved using the definition.
11. **Order-Preserving Function Rule:** if $h(n)$ is an order preserving function, and $f(n) \in \mathbf{O}(g(n))$, then $h(f(n)) \in \mathbf{O}h((g(n)))$, then
12. **FACT:** $1 \in \mathbf{O}(\lg n)$

The Master Theorem

The Master Theorem for finding closed form expressions for recurrence relations is expressed as follows. For a function $T(n)$ defined on positive integers, where $T(n) = aT(\frac{n}{b}) + f(n)$ where by $\frac{n}{b}$ we mean the floor or the ceiling of $\frac{n}{b}$, and $f(n)$ is a positive-valued function, and constants a and b are such that $a \geq 1$ and $b > 1$, then:

1. **If** $f(n)$ is $O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, **then** $T(n)$ is $\Theta(n^{\log_b a})$.

2. **If** $f(n)$ is $\Theta(n^{\log_b a})$ **then** $T(n)$ is $\Theta(n^{\log_b a} \log n)$

3. **If** $f(n) \in \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$, and there exists some $c, 0 < c < 1$ such that $af(\frac{n}{b}) < cf(n)$ when n is sufficiently large, **then** $T(n) \in \Theta(f(n))$.

Log Facts

1. By $\lg n$ we mean $\log_2 n$.
2. By $\log n$ we mean $\log_{10} n$.
3. $\log n \leq n \ \forall n \geq 1$.
4. $\log_a^d n$ is, by definition, $(\log_a n)^d$
5. $2^{\lg n} = n$
6. $\log_a n^b = b \log_a n$
7. $\log_b a = \frac{\log_c a}{\log_c b}$
8. $\log_b a = \frac{1}{\log_a b}$

9. $a^{\log_b c} = c^{\log_b a}$
10. $\log_c(a * b) = \log_c a + \log_c b$
11. $\log_4 5 = 1.161$
12. $\log_5 4 = 0.861$
13. $\log_4 3 = 0.792$
14. $\log_3 4 = 1.262$