\( f_1(n) \in O(g_1(n)) \)

\( f_2(n) \in O(g_2(n)) \)

\( f_1(n) \cdot f_2(n) = O(g_1(n) \cdot g_2(n)) \)

---

**FibonacciMultiply** \((X[0..n-1], Y[0..n-1])\):

- \( \text{hold} \leftarrow 0 \)
- for \( k \leftarrow 0 \) to \( n + m - 1 \):
  - for all \( i \) and \( j \) such that \( i + j = k \):
    - \( \text{hold} \leftarrow \text{hold} + X[i] \cdot Y[j] \in O(1) \)
  - \( Z[k] \leftarrow \text{hold} \mod 10 \leq O(1) \)
  - \( \text{hold} \leftarrow [\text{hold}/10] \leq O(1) \)

return \( Z[0..m+n-1] \)

**Total**: \( O(1 + n^2 + 1) \leq O(n^2) \)
\[ \Omega(g(n)) = \{ f(n) : \text{exist pos. constants } c, n_0 \text{ s.t.} \]
\[ 0 \leq c g(n) \leq f(n) \text{ for all } n \geq n_0 \} \]

\[ \text{for inner loop runs } \geq n/2 \in \Omega(n) \]

\[ \text{values of } k \text{ s.t.} \]

\[ \text{Mult has } \geq n/2 \in \Omega(n) \]
times,

\( \Rightarrow \) runs in \( \Omega(n^2) \) time

big-Theta: \( \Theta(g(n)) = \Omega(g(n)) \cap O(g(n)) \)

FibMult takes \( \Theta(n^2) \) time

"asymptotically tight bound"
Little o

\( o(g(n)) = \{ f(n) : \text{for all pos. constants } c, \text{there exists } n_0 \text{ s.t.} \)

\[ 0 \leq f(n) < c g(n) \text{ for all } n \geq n_0 \]
\[ 5n^2 = O(n^2) \]
\[ 20n^2 + O(n) = O(n^2) \]

"For all choices of functions in each bit of asymptotic notation on the left, there exist choices of function for right to make inequality true."
\[ F: \text{Mul} \]
\[ 0(1) + \sigma(n) \cdot (\sigma(1) + \sigma(n)) + \sigma(1) \leq O(n^2) \]
\[ 5n^2 + 1000n = 5n^2 + O(n) \]
\[ = O(n^2) \]
\[ = O(2^n) \]
\[ \leq O(2^{2^n}) \]
$f(n)$ is polynomially bounded if

\[ f(n) = O(n^k) \text{ for some constant } k. \]

\[ n^k = o(n^{k_2}) \iff k_1 < k_2 \]

most times from this running class
exponential functions

\[ n^k = o(a^n) \text{ for any constants } k + a > 1. \]

\[ a^n = o(c^n) \text{ for any } c > a > 1. \]

polylogarithmically bounded:

\[ (\log n) \in o(n^k) \text{ for any constants } 6 > 1, \ell, k > 0. \]

\[ \lg n := \log_2 n \]

\[ \ln n := \log_e n \]

\[ \log n := \log_{10} n \]
\[ \log_b n = \left( \log_b n \right)^2 \]

\[ \log_b n = \frac{\log_a n}{\log_a b} = \Theta \left( \log n \right) \]

(a and b constants)

\[ = \Theta \left( n \log n \right) = O \left( n \log n \right) \]

**WARNING:**

\[ \sum_{n=3}^{\infty} n \log_3 n = O \left( \sum_{n=1}^{\infty} \log n \right) \]
For running times

$O(n^2)$ is better than 25, $O(n^2)$

or $O(25n^2)$

$O(n^2) + O(n) = O(n^2)$

$O(5^{\log_3 n}) = O(n^{\log_3 5})$
A reduction from problem X to problem Y means an algorithm for X that uses an algorithm for Y as a "black-box" or subroutine.

Alg for X must be correct for any alg for Y.

(running time might care)
Math uses simple theorems called **lemmas**. Big important proof may reduce to already proven lemmas.
A theorem:
Let \( n \) be a positive integer.
A divisor is a positive integer \( p \) s.t. \( \frac{n}{p} \) is an integer.

\( n \) is prime if it has exactly two divisors, \( n + 1 \).
\( n \) is composite if it has \( > 2 \) divisors.
Thm: Every integer \( n > 1 \) has a prime divisor.

Proof (?): Suppose there is an \( n > 1 \) with no prime divisor.

\( n \) is its own divisor, so \( n \) is not prime.

So there is a divisor \( d \) s.t. \( 1 < d < n \).

By assumption, \( d \) is not prime.

So \( \exists d \), is a divisor of \( d \).
\[ \frac{n}{d_1} = \left( \frac{n}{d} \right), \quad \left( \frac{d}{d_1} \right) \text{ is an integer so } d' \text{ divides } n. \]

So \( d_1 \) is not prime...
Try 2: Assume Thm is wrong. Let \( n > 1 \) be the smallest counter example.

**n is not prime**

So there is a divisor \( 1 < d < n \).

By assumption, \( d \) has a prime divisor \( 1 < p \leq d \).

\[
\frac{n}{p} = \frac{n}{d} \cdot \frac{d}{p}
\]

is an integer, so \( p \) is a prime.
divisor of $n$. $1$
Proof: Let \( n > 1 \). Assume all \( k > 1 \) are prime, \( 1 < k < n \) has a prime divisor.

Suppose \( n \) is prime.
- \( n \) is its own prime divisor.

Suppose \( n \) is not prime.
- There is a divisor \( 1 < d < n \).
- By assumption, \( d \) has a prime divisor \( p \).
- \( p \) is a prime divisor of \( n \).
In all cases, \( n \) has a prime divisor. 

\[ \square \]

Proof by induction.