3. $2, \log_3 n, \log_2 n, n^{2/3}, 20^n, 4n^2, 3^n, n!$

4. a) $T(n) = 3 \cdot 2^n$

~~The~~ The constant value will be irrelevant as n grows, therefore

$$2^n \cdot 64 = 2^n \cdot 2^6 = 2^{n+6}$$

The new computer will compute n_0+6 values in time t.

b). $T(n) = n^2$

$$\sqrt{64n^2} = 8n$$

The new computer will compute $8n$ values in time t.

c) $T(n) = 8n$

Once again, the constant value will become irrelevant as n grows

$64n$

The new computer will compute $64n$ values in time t

5. 100 x faster for each

$n_0 \to 100 n_0$

$n_0^2 \to \sqrt{100 \cdot n_0^2} = 10 n_0$

$n_0^3 \to \sqrt[3]{100 n_0^3} = \sqrt[3]{100}\ n_0$

$2^n \to 100 \cdot 2^n = 2^{\log_2 100} \cdot 2^n = 2^{n + \log_2 100}$

$n_0 + \log_2 100$

6 a) $T(n) = x$

$T(2n) = x^2$

$T(4n) = (x^2)^2 = x^4$

$T(8n) = (x^4)^2 = x^8$

Growth rate: x^n, where x is the given base value

b) $T(n) = x \to x^{3^0}$

$T(2n) = x^3 \to x^{3^1}$

$T(4n) = (x^3)^3 = x^9 \to x^{3^2}$

$T(8n) = (x^9)^3 = x^{27} \to x^{3^3}$

Growth rate: $x^{3^{\log_2 n}}$, where x is the given base value

11. a) $f(n) = \log n^2 \quad g(n) = \log n+5$

$$\lim_{n\to\infty} \frac{\log n^2}{\log n+5} = \log_{n+5} n^2$$

$\to \infty$

$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$, so $f(n)$ is in $\Omega(g(n))$

b) $f(n) = \sqrt{n} \quad g(n) = \log n^2$

$$\lim_{n\to\infty} \frac{\sqrt{n}}{\log n^2}$$

$f(n)$ is in $\Omega(g(n))$

c) $f(n) = \log^2 n \quad g(n) = \log n$

$$\lim_{n\to\infty} \frac{\log^2 n}{\log n} = \frac{\log(\log n)}{\log n} \quad \frac{d}{dn}$$

$$\frac{\frac{1}{\log n} \cdot \cancel{\frac{1}{n}}}{\cancel{\frac{1}{n}}} = \frac{1}{\log n} \to 0$$

$f(n)$ is in $O(g(n))$

d) $f(n) = n \qquad g(n) = \log^2 n$

$f(n)$ is in $\Omega(g(n))$

e) $f(n) = n\log n + n \qquad g(n) = \log n$

$$\lim_{n\to\infty} \frac{n\log n + n}{\log n} \frac{d}{dn}$$

$$= \frac{n\frac{1}{n} + \log n + 1^{\nearrow\infty}}{\cancel{\frac{1}{n}}^{\nearrow 0}}$$

$f(n)$ is in $\Omega(g(n))$

f) $f(n) = \log n^2 \qquad g(n) = (\log n)^2$

$$\lim_{n\to\infty} \frac{\log n^2}{(\log n)^2} \frac{dn}{dn}$$

$$\frac{\frac{1}{n^2}\cdot 2n}{2\log n \cdot \frac{1}{n}} = \frac{\frac{2n}{n^2}}{\frac{2\log n}{n}} = \frac{\frac{2}{n}}{\frac{2\log n}{n}}$$

$$= \frac{2}{2\log n} = \frac{1}{\cancel{\log n}^{\nearrow\infty}} = 0$$

$f(n)$ is in $O(g(n))$

g) $f(n) = 10 \qquad g(n) = \log 10$

$$\lim_{n\to\infty} = \frac{10}{\log 10} = \frac{10}{1} = 10$$

$f(n) = \Theta(g(n))$

h) $f(n) = 2^n \qquad g(n) = 10n^2$

$$\lim_{n\to\infty} \frac{2^n}{10n^2} \frac{d}{dn} = \frac{2^n \cdot \log(2)}{20n} \frac{d}{dn}$$

$$= \frac{\cancel{2^n}^{\nearrow\infty}(\log 2)^2}{20} = \infty$$

$f(n)$ is in $\Omega(g(n))$

i) $f(n) = 2^n \qquad g(n) = n\log n$

$$\lim_{n\to\infty} \frac{2^n}{n\log n} \frac{d}{dn} = \frac{2^n \log 2}{n\frac{1}{n} + \log n}$$

$$= \frac{2^n \log 2}{\log n + 1} \frac{d}{dn}$$

$f(n)$ is in $\Omega(g(n))$ $= \frac{2^n (\log 2)^{2\nearrow\infty}}{\cancel{\frac{1}{n}}^{\nearrow 0}}$

j) $f(n) = 2^n \qquad g(n) = 3^n$

$$\lim_{n\to\infty} \frac{2^n}{3^n}$$

$$\lim_{n\to\infty} \cancel{\frac{2}{3}}^{\nearrow 0}$$

$f(n)$ is in $O(g(n))$

K) $f(n) = 2^n \qquad g(n) = n^n$

$$\lim_{n\to\infty} \frac{2^n}{n^n}$$

$f(n)$ is in $O(g(n))$

12.

a) Since the operations are both constant statements, ~~then~~

$\boxed{\Theta(1)}$

b) Outside : 1

Outer loop: 3

Inner loop: n

Inside the loop: 1

$\Theta(3n+1)) = \underline{\Theta(n)}$

c) Outside loop: 1

loop: n^2

Inside loop: 1

$\Theta(n^2+1+1) = \underline{\Theta(n^2)}$

d) Outer loop: n-1

Inner loop: n-1

Inside the loop: 1

$\Theta(n-1((n-1)-1)) = \underline{\Theta(n^2)}$

e) outside the loop: 1

Outer: n

Inner: logn

Inside the loop: 1

$\Theta(n(\log n+1)) = \underline{\Theta(n \log n)}$

f) outside the loop: 1

Outer loop: logn

inner loop: n

inside the loop: 1

$\Theta(\log n(n+1)+1) = \underline{\Theta(n \log n)}$

g) Outer loop: n

inner loop: n

Inside the loop: n logn

$\Theta(n(n \log n)) = \underline{\Theta(n^2 \log n)}$

h) outside the loop: 1

Outer loop: n

inner loop: n/2

inside the loop: 1

$\Theta(n(n/2(1)) = \frac{n^2}{2} = \underline{\Theta(n^2)}$

i) outside the loop: 1

if: $n(1) = n$

else: 1

the if statement will only happen half the time, assuming n starts as a random int valu

So, half the time, the program will be $\Theta(n)$ and the other half it will be $\Theta(1)$.

We must take $\underline{\Theta(n)}$, because it is larger