Exercise 7.1 Going back to Example 7.5, suppose you are trying out different pairs of constants C and k that fit the Big-$\mathcal{O}$ definition (the ones we picked there, $C = 20$ and $k = 1\,000$, are not the only ones we could have picked; in fact, an infinite number of such pairs fit the definition.)

Suppose you try $C = 5$. Is there a value of k such that $T(n) \leq 5n$ for all $n > k$?

$T(n) = 10n + 10000$

Assume such k exists that $T(n)$ is $O(g(n))$ with $C = 5$ as a witness.

From the definition of Big-O notation, we obtain:

$10n + 10000 \leq 5n$ for all $n > k$

$\Rightarrow 5n \leq -10000$

$\Rightarrow n \leq -2000$

This is invalid since n (the size of a program's input) cannot be negative.

Therefore, there exists no such k that satisfies $T(n)$ is $O(g(n))$ with $C = 5$.

Exercise 7.2 Use the Big-$\mathcal{O}$ definition to show that 2 is in $O(1)$ and 3 is in $O(n)$.

(a) We need to show that 2 is in $O(1)$ by finding C, k to fit Big-O definition.

Take $C = 4$ and $k = 0$, we obtain: $2 \leq 4(1)$ for all $n \geq 0$ which is always true

This is the same as $2 \leq C.(1)$ for all $n \geq k$ with $C = 4$, $k = 0$ as witnesses.

Therefore, 2 is $O(1)$

(b) We need to show that 3 is in $O(n)$ by finding C, k to fit Big-O definition.

Take $C = 3$ and $k = 1$, we obtain: $1 \leq n \Rightarrow 3 \leq 3.n$ for all $n \geq 1$

which means $3 \leq C.n$ for all $n \geq k$ with $C = 3$, $k = 1$ as witnesses.

Therefore, 3 is $O(n)$.

Exercise 7.3 At the recent annual (and glitzy) event, in front of a live, captive audience, Steve Gates proudly presents MicroAppleSoft's latest computer. He executes a program on an input of size n that takes 1 minute to complete.
FastAndLoose Hardware Corp. representative claims that their newest computer will run 100 times faster than MicroAppleSoft's latest machine.

What size input can FastAndLoose's computer execute in one minute for each algorithm with the following growth rates?

(a) n (b) n^2 (c) n^3 (d) 2^n

(a) $100n$

(b) $100n^2 \longrightarrow$ size input $= 10n$

(c) $100n^3 \longrightarrow$ size input $= \sqrt[3]{100n^3} \approx 4.64n \approx 5n$

(d) size input $= x \rightarrow 2^x = 100.2^n$

$\rightarrow x = \log(100.2^n) = \log 10^2 + n\log 2 = 2\log 10 + n\log 2$

Exercise 7.4 Arrange the following functions by their order of growth from the slowest to the fastest.

$5n^4 \quad 3^n \quad \log_2 n^6 \quad 7n \quad 89\log_2 n \quad 2n\log_2 n^3 \quad n^{8/3}$

$89\log_2 n < \log_2 n^6 < 7n < 2n\log_2 n^3 < n^{8/3} < 5n^4 < 3^n$

Exercise 7.5 Show that n^2 is not in $O(n)$.

Suppose that n^2 is $O(n)$ then by Big-O definition, there is such C and k that:

$n^2 \leq C.n$ for all $n \geq k \Rightarrow n \leq C$ for all $n \geq k$

This cannot be true no matter how large the constant C is because n can be arbitrarily large.

It follows that no such C, k exist that meet the Big-O definition, showing that n^2 is not $O(n)$.

Exercise 7.6 Let $T(n) = 1 + 2 + 3 + \ldots + n$.

$$T(n) = \frac{n(n+1)}{2} = \frac{n^2+n}{2}$$

(a) Is it true that $T(n)$ is $O(n^3)$?

(b) Is it true that $T(n)$ is $O(n)$?

(c) Is it true that $T(n)$ is $O(n^2)$?

(d) Is it true that $T(n)$ is $\Theta(n^2)$?

(a) True $\longrightarrow$ Take $k = 1$: $1 \leq n \rightarrow n^2 \leq n^3$, $n \leq n^3$ $\Rightarrow n^2 + n \leq 2n^3 \Rightarrow \frac{n^2+n}{2} \leq 1.n^3$ for $n \geq 1 \Rightarrow T(n)$ is $O(n^3)$ with $(C, k) = (1, 1)$ as witnesses

(b) False $\longrightarrow$ Suppose that $T(n)$ is $O(n)$, by the Big-O definition, we obtain: $\frac{n^2+n}{2} \leq C.n \Rightarrow n(n+1) \leq 2C.n \Rightarrow n+1 \leq 2C \Rightarrow n \leq 2C-1$

This cannot be true no matter how large the constant C is because n can be arbitrarily large. It follows that no such C, k exist that meet the definition of Big-O, showing that $T(n)$ is not $O(n)$.

(c) True $\longrightarrow$ Take $k = 1$: $\left.\begin{array}{l} 1 \leq n \Rightarrow n \leq n^2 \\ n^2 \leq n^2 \end{array}\right\} \Rightarrow n^2 + n \leq 2n^2 \Rightarrow \frac{n^2+n}{2} \leq 1.n^2$ for $n \geq 1 \Rightarrow T(n)$ is $O(n^2)$ with $(C, k) = (1, 1)$ as witnesses.

(d) True $\longrightarrow$ Take $k = 1$: $\left.\begin{array}{l} 1 \leq n \Rightarrow 0 \leq 4n \\ n^2 \leq n^2 \Rightarrow 2n^2 \leq 4n^2 \end{array}\right\} \Rightarrow 2n^2 \leq 4n^2 + 4n \Rightarrow n^2 \leq 2n^2 + 2n = 2(n^2+n) = 4\left(\frac{n^2+n}{2}\right) \Rightarrow n^2$ is $O(T(n))$ with $(C, k) = (4, 1)$ as witnesses.

Since $T(n)$ is $O(n^2)$ (part c) and n^2 is $O(T(n))$, $T(n)$ is $\Theta(n^2)$

Exercise 7.7 Select all true statements from the following choices:

✓ (a) $2168n^2 + 21n + 68$ is $O\left(\frac{1}{21^{68}}n^3 + 2n + 1\right)$.

(b) $\frac{1}{21^{68}}n^3 + 2n + 1$ is $O(2168n^2 + 21n + 68)$.

✓ (c) $\frac{1}{21^{68}}n^3 + 2n + 1$ is not $O(2168n^2 + 21n + 68)$.

(d) $2168n^2 + 21n + 68$ is not $O\left(\frac{1}{21^{68}}n^3 + 2n + 1\right)$.

True statements based on this theorem: If $d > c > 1$, then n^c is $O(n^d)$

Exercise 7.8 True or False:

(a) $(n+1)(n-1)$ is $O(\log n^2) \longrightarrow$ False

(b) $(n+1)(n-1)$ is $O(\log n)^2 \longrightarrow$ False

(a) $(n+1)(n-1) = n^2 - 1 \leq n^2$

$\log n^2 = 2\log n \leq 2n \leq n^2$

Since $(n+1)(n-1)$ is $O(n^2)$ but $n^2 \geq \log n^2$, the statement is false.

(b) $\log n \leq n \Rightarrow (\log n)^2 \leq n^2$

The statement is false.

Exercise 7.9 What does the following strangely named method do and what is its running time?

```
void esrever(int[] a) {
   for (int here = 0; here < a.length / 2; here++) {
      int there = a.length - here - 1;
      int hold = a[here];
      a[here] = a[there];
      a[there] = hold;
   }
}
```

- The method reverses an order of the array
- Running time: $O(n)$

Exercise 7.10 You are given the following functions:

$f_1(n) = 2n + n$, $f_2(n) = 2^{\log_2 n}$, and $f_3(n) = (n+1)(n-1)$.

$= 3n$ $\quad = n$ $\quad = n^2 - 1$

Which of the following is true? (Choose all that apply.)

✓ (a) $f_1(n)$ is $O(f_3(n))$.

✓ (b) $f_2(n)$ is $O(f_3(n))$.

(c) $f_3(n)$ is $O(f_1(n))$.

✓ (d) $f_2(n)$ is $O(f_1(n))$.

✓ (e) $f_1(n)$ is $O(f_2(n))$.

(f) $f_3(n)$ is $O(f_2(n))$.

True	False	Explanation
(a)		$3n \leq n^2 - 1 \Rightarrow 3n$ is $O(n^2 - 1)$
(b)		$n \leq n^2 - 1 \Rightarrow n$ is $O(n^2 - 1)$
	(c)	$n^2 - 1 \geq 3n \Rightarrow n^2 - 1$ is not $O(3n)$
(d)		$n \leq 3n \Rightarrow n$ is $O(3n)$
(e)		$3n \leq 4.n \Rightarrow 3n$ is $O(n)$ $(C, k) = (4, 1)$
	(f)	$n^2 - 1 \geq n \Rightarrow n^2 - 1$ is not $O(n)$

Exercise 7.11 You are given the following functions:

$f_1(n) = \sqrt{n}$, $f_2(n) = 3^n$, and $f_3(n) = n^3$.

Which of the following is true? (Choose all that apply.)

✓ (a) $f_1(n)$ is $O(f_3(n))$. (d) $f_2(n)$ is $O(f_1(n))$.

(b) $f_2(n)$ is $O(f_3(n))$. ✓ (e) $f_1(n)$ is $O(f_2(n))$.

(c) $f_3(n)$ is $O(f_1(n))$. ✓ (f) $f_3(n)$ is $O(f_2(n))$.

(a) True $\longrightarrow \sqrt{n} = n^{1/2} \leq n^3 \Rightarrow \sqrt{n}$ is $O(n^3)$

(b) False $\longrightarrow 3^n \geq n^3 \Rightarrow 3^n$ is not $O(n^3)$

(c) False $\longrightarrow n^3 \geq \sqrt{n} \Rightarrow n^3$ is not $O(\sqrt{n})$

(d) False $\longrightarrow 3^n \geq \sqrt{n} \Rightarrow 3^n$ is not $O(\sqrt{n})$

(e) True $\longrightarrow \sqrt{n} \leq 3^n \Rightarrow \sqrt{n}$ is $O(3^n)$

(f) True $\longrightarrow n^3 \leq 3^n \Rightarrow n^3$ is $O(3^n)$

Exercise 7.12 Arrange the functions

$8\sqrt{n} \quad (\log n)^2 \quad 2n\log n \quad n! \quad (1.1)^n \quad n^2$

in a list so that each function is Big-O of the next function.

$(\log n)^2 < 8\sqrt{n} < 2n\log n < n^2 < (1.1)^n < n!$