

## 6.x — Chapter 6 summary and quiz

 **ALEX**  **SEPTEMBER 11, 2023**

### Quick review

Always use parentheses to disambiguate the precedence of operators if there is any question or opportunity for confusion.

The arithmetic operators all work like they do in normal mathematics. The remainder (%) operator returns the remainder from an integer division.

The increment and decrement operators can be used to easily increment or decrement numbers. Avoid the postfix versions of these operators whenever possible.

Beware of side effects, particularly when it comes to the order that function parameters are evaluated. Do not use a variable that has a side effect applied more than once in a given statement.

The comma operator can compress multiple statements into one. Writing the statements separately is usually better.



The conditional operator is a nice short version of an if-statement, but don't use it as an alternative to an if-statement. Only use the conditional operator if you use its result.

Relational operators can be used to compare floating point numbers. Beware using equality and inequality on floating point numbers.

Logical operators allow us to form compound conditional statements.

---

### Quiz time

#### Question #1

Evaluate the following:

a) `(5 > 3 && 4 < 8)`

[Show Solution \(javascript:void\(0\)\)](#)

b) `(4 > 6 && true)`

[Show Solution \(javascript:void\(0\)\)](#)

c) `(3 >= 3 || false)`

[Show Solution \(javascript:void\(0\)\)](#)

d) `(true || false) ? 4 : 5`

[Show Solution \(javascript:void\(0\)\)](#)

---

#### Question #2

Evaluate the following:

a) 7 / 4

[Show Solution \(javascript:void\(0\)\)](#)

b) 14 % 5

[Show Solution \(javascript:void\(0\)\)](#)

Question #3

Why should you never do the following:

a) `int y{ foo(++x, x) };`

[Show Solution \(javascript:void\(0\)\)](#)

b) `double x{ 0.1 + 0.1 + 0.1 }; return (x == 0.3);`

[Show Solution \(javascript:void\(0\)\)](#)

c) `int x{ 3 / 0 };`

[Show Solution \(javascript:void\(0\)\)](#)



**Next lesson**

**0.1** [Bit flags and bit manipulation via std::bitset](#)



**[Back to table of contents](#)**



**Previous lesson**

**6.7** [Logical operators](#)

...



Leave a comment...



Name\*



Email\*



Notify me about replies:



**POST COMMENT**



Find a mistake? Leave a comment above!?



Avatars from <https://gravatar.com/> are connected to your provided email address.

We and our partners share information on your use of this website to help improve your experience.



Do not sell my info: ☐

OKAY