

The List of Operators

The first operators are the arithmetic operators, which are very similar to almost every other programming language. When you write the cards, the description side should say that it's an arithmetic operator, and what it does.

Arithmetic Operators	
Operator	Description
+	Add
-	Subtract
*	Multiply
/	Divide
%	Modulus
++	Increment
--	Decrement

Relational operators test values for equality, and again, they are very common in programming languages.

Relational Operators	
Operator	Description
==	Equal
!=	Not equal
>	Greater than
<	Less than
>=	Greater than equal
<=	Less than equal

Logical operators perform logic tests, and you should already know what these do. The only odd one is the *logical ternary*, which you'll learn later in this book.

Logical Operators	
Operator	Description
&&	Logical and
	Logical or
!	Logical not
? :	Logical ternary

Bitwise operators do something you likely won't experience often in modern code. They alter the bits that make up bytes and other data types in various ways. I won't cover this in my book, but they are very handy when working with certain types of lower-level systems.

Bitwise Operators	
Operator	Description
&	Bitwise and
	Bitwise or
^	Bitwise xor
~	Bitwise one's complement
<<	Bitwise shift left
>>	Bitwise shift right

Assignment operators simply assign expressions to variables, but C combines a large number of other operators with assignment. So when I say `and-equal`, I mean the *bitwise* operators, not the logical operators.

Assignment Operators	
Operator	Description
=	Assign equal
+=	Assign plus-equal
-=	Assign minus-equal
*=	Assign multiply-equal
/=	Assign divide-equal
%=	Assign modulus-equal
<<=	Assign shift-left-equal
>>=	Assign shift-right-equal
&=	Assign and-equal
^=	Assign xor-equal
=	Assign or-equal

I'm calling these *data operators* but they really deal with aspects of pointers, member access, and various elements of data structures in C.

Data Operators	
Operator	Description
sizeof()	Get the size of
[]	Array subscript
&	The address of
*	The value of
->	Structure dereference
.	Structure reference

Finally, there are a few miscellaneous symbols that are either frequently used for different roles (like `,`), or don't fit into any of the previous categories for various reasons.

Assignment Operators	
Operator	Description
=	Assign equal
+=	Assign plus-equal
-=	Assign minus-equal
*=	Assign multiply-equal
/=	Assign divide-equal
%=	Assign modulus-equal
<<=	Assign shift-left-equal
>>=	Assign shift-right-equal
&=	Assign and-equal
^=	Assign xor-equal
=	Assign or-equal

I'm calling these *data operators* but they really deal with aspects of pointers, member access, and various elements of data structures in C.

Data Operators	
Operator	Description
sizeof()	Get the size of
[]	Array subscript
&	The address of
*	The value of
->	Structure dereference
.	Structure reference

Finally, there are a few miscellaneous symbols that are either frequently used for different roles (like `,`), or don't fit into any of the previous categories for various reasons.

Miscellaneous Operators	
Operator	Description
,	Comma
()	Parentheses
{ }	Braces
:	Colon
//	Single-line comment start
/*	Multi-line comment start
*/	Multi-line comment end

Study your flash cards while you continue with the book. If you spent 15–30 minutes a day before studying, and another 15–30 minutes before bed, you could most likely memorize all of these in a few weeks.