

CIS241

System-Level Programming and Utilities

C Conditionals and Operations

Erik Fredericks, frederer@gvsu.edu

Fall 2025

Based on material provided by Erin Carrier, Austin Ferguson, and Katherine Bowers

Basic operators:

- Assignment: `int x = 7;`
- Arithmetic:
 - Basics: `+`, `-`, `/`, `*`
 - Modulo: `%` (Remainder)
 - Modified assignment (do op and store):
 - `+=`, `-=`, `*=`, `/=`, `%=`
 - Increment / decrement: `++`, `--`

Boolean operators

- **Comparison:** `==`, `!=`, `>`, `<`, `>=`, `<=`
- **Boolean operations:**
 - Not: `!`
 - And: `&&`
 - Or: `||`
- **Note: These are different than Boolean operators:**
 - Not (inverse): `~`
 - Bitwise and: `&`
 - Bitwise or: `|`
 - Bitwise xor: `^`

Control statements

```
if (expr)
{
    statement1;
}
else if (expr2)
{
    statement2;
}
else
{
    statement3;
}

// single line statements don't need braces
if (expr3) statement;
```

Ternary operator

```
expression ? expression1 : expression2
```

- if `expression` is true, replace with `expression1`
else, replace with `expression2`

Order of operations

https://en.cppreference.com/w/c/language/operator_precedence

Precedence	Operator	Description	Associativity
1	++ --	Suffix/postfix increment and decrement	Left-to-right
	()	Function call	
	[]	Array subscripting	
	.	Structure and union member access	
	->	Structure and union member access through pointer	
	(type){ list }	Compound literal(c99)	
2	++ --	Prefix increment and decrement ^[note 1]	Right-to-left
	+ -	Unary plus and minus	
	! ~	Logical NOT and bitwise NOT	
	(type)	Cast	
	*	Indirection (dereference)	
	&	Address-of	
	sizeof	Size-of ^[note 2]	
	_Alignof	Alignment requirement(c11)	
3	* / %	Multiplication, division, and remainder	Left-to-right
4	+ -	Addition and subtraction	
5	<< >>	Bitwise left shift and right shift	
6	< <=	For relational operators < and ≤ respectively	
	> >=	For relational operators > and ≥ respectively	
7	== !=	For relational = and ≠ respectively	
8	&	Bitwise AND	
9	^	Bitwise XOR (exclusive or)	
10		Bitwise OR (inclusive or)	
11	&&	Logical AND	
12		Logical OR	
13	?:	Ternary conditional ^[note 3]	Right-to-left
14 ^[note 4]	=	Simple assignment	
	+= -=	Assignment by sum and difference	
	*= /= %=	Assignment by product, quotient, and remainder	
	<<= >>=	Assignment by bitwise left shift and right shift	
	&= ^= =	Assignment by bitwise AND, XOR, and OR	
15	,	Comma	Left-to-right