

Introduction to Modern C++ Course Outline

Ryan Baker

January 2, 2025

Week 1: Introduction to C++

1	Course Introduction	3
1.1	Overview of Lecture Series	3
2	Features of C++	3
2.1	Evolution of C++	3
2.2	The C++ Methodology	3
2.3	C++ vs. Other Languages	3
3	Environment Setup	3
3.1	Tools Required	3
3.1.1	Text Editor	3
3.1.2	Compiler	3
3.2	“Hello, World!” Example	3
4	Basic Syntax and Structure	3
4.1	Basic Structure of a C++ Program	3
4.1.1	<code>int</code> <code>main()</code>	3
4.2	Foundational Concepts	3
4.2.1	Semicolons, <code>/* comments */</code> , and Whitespace	3
4.2.2	Line-by-Line Execution	3
4.3	Input and Output	3
5	Datatypes and Variables	3
5.1	Primitive Types	3
5.1.1	<code>int</code> Type	3
5.1.2	<code>char</code> Type	3
5.1.3	<code>bool</code> Type	3
5.1.4	<code>float</code> Type	3
5.1.5	<code>void</code> Type	3
5.2	Declaration and Definition	3
5.2.1	Assignment Operator <code>=</code>	3
5.2.2	Brace Initialization <code>{}</code>	3
5.3	Arithmetic Operators	3

Week 2: How C++ Works

1	The Build Process	2
1.1	Source Code	2
1.2	Preprocessor	2
1.2.1	Text Substitution	2
1.2.2	Conditional Compilation	2
1.2.3	File Inclusion	2
1.2.4	Preprocessor Output	2
1.3	Compilation	2
1.3.1	Compiler Output	2
1.4	Linking	2
2	Introduction to Memory	2
2.1	How C++ Uses Memory	2
2.2	Pointers	2
2.2.1	NULL Pointers	2
2.2.2	Pointer Arithmetic	2
2.2.3	Pointers to Pointers	2
3	Memory Layout	2
3.1	Text Segment	2
3.2	Static Memory	2
3.2.1	Variable Lifetime	2
3.3	The Heap	2
3.3.1	Operators new and delete	2
3.3.2	Memory Leaks	2
3.4	The Stack	2
3.4.1	The Stack Pointer	2

Week 3: C++ Control Flow

1	Functions	2
2	Scope	2
3	Conditions and Branching	2
4	Loops	2
5	Control Flow Keywords	2

1	Arrays	2
2	Structs	2
3	Classes	2
1	Principles of Object-Oriented Programming	2
2	Operator Overloading	2
1	Introduction to the Standard Library	2
2	Standard Containers	2
3	Standard Algorithms	2
4	Smart Pointers	2
1	Function Templates	2
2	Class Templates	2
3	Template Specialization	2
4	Variadic Templates	2
1	Lambdas	2
2	Compile-Time Programming	2
3	Template Metaprogramming	2