Programming Tutorials - C, C++, OpenGL, STL

Welcome! If you're new to C++, I recommend you purchase my ebook, Jumping into C++, a complete step-by-step guide for beginners.

If you're looking for free tutorials, learn C++ with our C++ tutorial, starting at C++ Made Easy, Lesson 1 (all lessons)

If you want to learn C instead, check out our C tutorial C Made Easy, Lesson 1 (all lessons)

Want more advanced material on C, C++ graphics, game programming or algorithms? We've got that too.

List advanced tutorial topics »

C++ Tutorial, C++ Made Easy: Learning to Program in C++

Learn C++ with this tutorial, designed for beginners and containing lots of examples, tips and simple explanations.

- Intro to C++ (Quiz)
- If statements (Quiz)
- Loops in C++ (Quiz)
- Functions in C++ (Quiz)
- Switch case (Quiz)
- Accessing Memory with Pointers (Quiz)
- Structures in C++ (Quiz)
- Storing data with Arrays (Quiz)
- Character Strings in C++ (Quiz)
- File I/O (Quiz)
- Typecasting (Quiz)
- Classes and introduction to object-oriented programming (Quiz)
- Inline functions (Quiz)
- Command line arguments (Quiz)
- Linked Lists
- Recursion
- Variable argument lists for functions
- Binary Trees
- Overview of Inheritance
- Inheritance Syntax and Examples
- C++ Class Design
- Enumerated types
- Formatted Output in C++ using ismanip
- Templates in C++
- Initialization Lists and Inheritance
- Templated functions
- Template specialization and partial specialization
- Understanding the C Preprocessor -- Constants, Macros, and other Tricks
- Generating random Numbers
- Using Modulus to get remainders

C Tutorial - C Made Easy

This tutorial is based on the above tutorial, but uses only standard C language features.

- Intro to C (Quiz)
- If statements (Quiz)
- Loops in C (Quiz)
- Functions and Program Organization (Quiz)
- Switch case (Quiz)
- Pointers (Quiz)
- Structures
- Arrays (Quiz)
- C-style Strings (Quiz)
- C File I/O
- Typecasting (Quiz)
- Command line arguments (Quiz)
- Linked Lists
- Recursion
- Variable argument lists
- Binary Trees

More Advanced C and C++ Language Feature Tutorials

- Bitwise Operators in C and C++
- File I/O in C
- Print format strings for beautiful output
- What's the difference between declaring and defining something in C and C++?
  Learn about the distinction between declaring a variable, class or function--and defining it--and why it matters when you have trouble compiling your code
- Removing the Mystery of Function Pointers
- Functors: Function Objects in C++
- References in C++
- Const Correctness--Why bother?
- How and When to Use Private Inheritance
- When To Use Goto: A Partial Defense of the goto Statement
- Using Exceptions for Error Handling in C++
- Organizing programs with Namespaces in C++
- Understanding the Static Keyword in C and C++
  A multipurpose keyword, static can be used globally, locally, and on class variables, with different meanings in each context
- Templated Classes in C++
- Using auto _ptr to avoid memory leaks in C++
- Understanding std::string (C++ strings)
- The friend keyword and data encapsulation
- Secure Coding - Preventing Buffer Overflows and other attacks
- Unicode: What you Can Do About it Today
- Multiple Inheritance in C++
- Solving the Diamond Problem with Virtual Inheritance
- Operator overloading
- Understanding Constructors, Destructors and Object Lifetime
- A gentle introduction to C++ I/O streams

[C++11 - the new C++ standard]

C++11 is the new C++ standard, and it's chock full of goodness for C++ programmers, old and new.

- What is C++11? - Get introduced to the new C++ standard, C++11 (previously known as C++0x)
- Auto, Declsype and return values after functions in C++11
- Lambda Functions in C++11 - the definitive guide
- Range-Based For Loops in C++11
- Generalized Constant Expressions in C++11 with constexpr
- Rvalue References and Move Semantics in C++11
- Enum classes and nullptr in C++11

[C++ Standard Template Library (STL) tutorials]

The STL is a powerful library that comes as part of standard C++, and should be a tool used and understood by all C++ programmers.

- An Introduction to the Standard Template Library (STL)
- The STL Vector Container
- STL Iterators
- STL Map Container
- STL List Container

Understanding Floating Point Numbers

by Jeff Bezanson

Every programmer should understand enough about floating point numbers to avoid the pitfalls of assuming perfect precision.

- Background: Accuracy vs. Precision
- Floating Point Representation
- Bonus: Printing Floating Point Numbers Cleanly

Using Remote Procedure Calls (RPCs)

by Jeff Bezanson

- Background and Using the RPC compiler
- Writing the Server Code
- Writing the Client and Putting it All Together
- Get the code

Same Game - A Simple Game from Start to Finish

by Ben Marchant

Do you want to learn how to create a game? This series will teach you how to create a game, starting from the very beginning and ending with a fully playable game.

- Same Game - Part 1: Introduction to technologies and drawing the game board
- Same Game - Part 2: Creating a real, playable game
- Same Game - Part 3: Adding difficulty levels and other menu options
- Same Game - Part 4: Changing the game board size and the block count
- Same Game - Part 5: Adding undo/redo functionality and keyboard accelerators

Object-Oriented Animation

Learn how to design a framework for doing animation—a foundation for many games and graphics engines.

- An Introduction to Object Oriented Animation
- Frames, Layers and Layer Folders
- The design of the animation engine

Using Microsoft ADO with SQL Databases in C++

by Patrick Mancier

- Overview of this tutorial
- Part 1: Introduction to SQL
- Part 2: Introduction to ADO
- Part 3: ADO Wrapper Classes
- Part 4: Creating the ADO Manager Class
- Part 5: Using the CADOManager in Practice

SDL Tutorials

The SDL is a simple library for doing graphics in C and C++

- Setting up SDL
OpenGL Tutorials

By RoD

- Intro to OpenGL
- OpenGL vs DirectX
- OpenGL and Windows
- The WinMain procedure
- Windows Programming
- Intro to WGL, the Windows Graphics Layer
- Your first OpenGL Program
- Projections in OpenGL

Rotations in Three Dimensions

By Confuted and Silversonic

This series of tutorials describes the math required for performing three dimensional rotations.

- The Basics of 3D rotations
- Rotation Matrices
- Rotation about an Arbitrary Axis
- Uses for what you’ve learned
- Using Quaternions

AI Tutorials

Learn about AI, including how to make game AI using the minimax algorithm.

- Perceptrons, a simple neuron simulator
- MiniMax Game Trees
- Chess Board Representation
- Solving problems with genetic algorithms

Data Structures

All programmers should know something about basic data structures like stacks, queues and heaps. Graphs are a tremendously useful concept, and two-three trees solve a lot of problems inherent in more basic binary trees.

- Stack Data Structure
- The Queue Data Structure
- Heaps
- Hash Tables
- Graphs in computer science
- Two-three trees

Algorithmic Efficiency and Sorting and Searching Algorithms

Learn how to determine the efficiency of your program and all about the various algorithms for sorting and searching--both common problems when programming.

- Algorithmic Efficiency and Big-O notation
- Efficiency and the space-time tradeoff
- Search Algorithms - linear search and binary search
- Comparison of Sorting Algorithms
- Intro to sorting algorithms: bubble sort
- Selection sort and Insertion sort
- Heap Sort
- Merge Sort
- Quicksort
- Radix Sort a special case sorting algorithm

Advanced Algorithms

If you’ve mastered the basics, perhaps you’d like to move to more advanced, specialized algorithms

- Exclusive-OR (XOR) Encryption
- Dijkstra’s Algorithm for finding shortest paths in graphs
- Dynamic Programming with an example of all pairs shortest paths
- Minimum Spanning Trees and Prim’s Algorithm
- Huffman Encoding Compression Algorithm

Computer Science Theory

If you’ve moved on from the advanced algorithms, perhaps you’d like to learn more about the fundamental nature of computation--a deep and rewarding topic.

- An introduction to the topic of computer science
- Base Systems - Binary, Hex and Octal
- The Halting Problem
  - Not for the faint of heart (or beginners), this tutorial covers an advanced topic in computer science: the nature of what can and cannot be computed -- what types of problems computers simply cannot solve
- What We Cannot Know: Consequences of the Halting Problem

Coding Style

- Programming Style, Part 1: Whitespace
- Programming Style, Part 2: Naming Conventions
- Programming Style, Part 3: How you can write readable code, and why you should
- How -- and Why -- to Comment
Compilers and Makefiles

- How to create a shared library on Linux using GCC
- Unravelling the mysteries of makefiles
- Advanced makefile tricks
- Dealing with the compilation process
- Why you should fix compiler warnings

Memory Management

- Using auto_ptr to manage memory
- Dynamic Memory Allocation, Part 1: new, delete and multidimensional array allocation
- Dynamic Memory Allocation, Part 2: Virtual Memory
- Dynamic Memory Allocation, Part 3: Overloading Operator New and Delete
- Dynamic Memory Allocation, Part 4: Common Memory Management Problems

Once you’re done with these tutorials, take the C++ Memory Management Quiz

Other Programming Languages

Interested in other programming languages, go beyond C and C++ to learn about Java, C# and more!

- C and C++ for Java Programmers
- Java for C++ Programmers: A Syntax Cheat Sheet
- A Summary of C#
- A Programming Language Comparison

Miscellaneous Programming Articles

- Test Driven Development, The Philosophy
- Where C and C++ Differ
- So you want write a game?
- How do you tell how large an object is?
- How to Start Writing a Program
- Module Development for Game Design
- Debugging binary search: the difficulty of getting your code right the first time
- New Year’s Resolutions for C/C++ Programmers
- 5 New Year’s Resolutions for Programmers, updated for 2011
- Thinking about Programming
- Common Coding Mistakes
- The 5 Most Common Problems New Programmers Face—And How You Can Solve Them
- The Secret to Learning Anything
- 5 Ways You can Learn Programming Faster