

uCertify

Course Outline

Introduction to C++



18 May 2024

1. Pre-Assessment
2. Exercises, Quizzes, Flashcards & Glossary
Number of Questions
3. Expert Instructor-Led Training
4. ADA Compliant & JAWS Compatible Platform
5. State of the Art Educator Tools
6. Award Winning Learning Platform (LMS)
7. Chapter & Lessons

Syllabus

Chapter 1: Introduction

Chapter 2: Hour 1: Writing Your First Program

Chapter 3: Hour 2: Organizing the Parts of a Program

Chapter 4: Hour 3: Creating Variables and Constants

Chapter 5: Hour 4: Using Expressions, Statements, and Operators

Chapter 6: Hour 5: Calling Functions

Chapter 7: Hour 6: Controlling the Flow of a Program

Chapter 8: Hour 7: Storing Information in Arrays and Strings

Chapter 9: Hour 8: Creating Basic Classes

Chapter 10: Hour 9: Moving into Advanced Classes

Chapter 11: Hour 10: Creating Pointers

Chapter 12: Hour 11: Developing Advanced Pointers

Chapter 13: Hour 12: Creating References

Chapter 14: Hour 13: Developing Advanced References and Pointers

Chapter 15: Hour 14: Calling Advanced Functions

Chapter 16: Hour 15: Using Operator Overloading

Chapter 17: Hour 16: Extending Classes with Inheritance

Chapter 18: Hour 17: Using Polymorphism and Derived Classes

Chapter 19: Hour 18: Making Use of Advanced Polymorphism

Chapter 20: Hour 19: Storing Information in Linked Lists

Chapter 21: Hour 20: Using Special Classes, Functions, and Pointers

Chapter 22: Hour 21: Using New Features of C++23

Chapter 23: Hour 22: Employing Object-Oriented Analysis and Design

Chapter 24: Hour 23: Creating Templates

Chapter 25: Hour 24: Dealing with Exceptions and Error Handling

Chapter 26: Appendix A: Binary and Hexadecimal

Chapter 27: Appendix B: Using the MinGW C++ Compiler on Windows

Videos and How To

8. Practice Test

Here's what you get

Features

9. Performance Based labs

Lab Tasks

Here's what you get

10. Post-Assessment

1. Pre-Assessment

Pre-Assessment lets you identify the areas for improvement before you start your prep. It determines what students know about a topic before it is taught and identifies areas for improvement with question assessment before beginning the course.

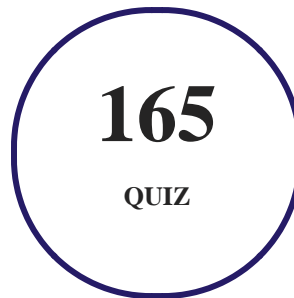
2. Exercises

There is no limit to the number of times learners can attempt these. Exercises come with detailed remediation, which ensures that learners are confident on the topic before proceeding.



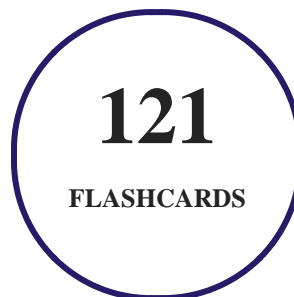
3. Quiz

Quizzes test your knowledge on the topics of the exam when you go through the course material. There is no limit to the number of times you can attempt it.



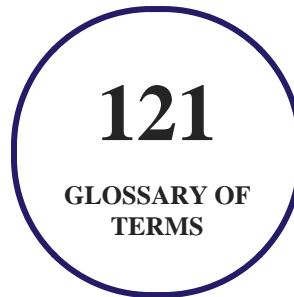
4. flashcards

Flashcards are effective memory-aiding tools that help you learn complex topics easily. The flashcard will help you in memorizing definitions, terminologies, key concepts, and more. There is no limit to the number of times learners can attempt these. Flashcards help master the key concepts.



5. Glossary of terms

uCertify provides detailed explanations of concepts relevant to the course through Glossary. It contains a list of frequently used terminologies along with its detailed explanation. Glossary defines the key terms.



6. Expert Instructor-Led Training

uCertify uses the content from the finest publishers and only the IT industry's finest instructors. They have a minimum of 15 years real-world experience and are subject matter experts in their fields. Unlike a live class, you can study at your own pace. This creates a personal learning experience and gives you all the benefit of hands-on training with the flexibility of doing it around your schedule 24/7.

7. ADA Compliant & JAWS Compatible Platform

uCertify course and labs are ADA (Americans with Disability Act) compliant. It is now more accessible to students with features such as:

- Change the font, size, and color of the content of the course
- Text-to-speech, reads the text into spoken words
- Interactive videos, how-tos videos come with transcripts and voice-over
- Interactive transcripts, each word is clickable. Students can clip a specific part of the video by clicking on a word or a portion of the text.

JAWS (Job Access with Speech) is a computer screen reader program for Microsoft Windows that reads the screen either with a text-to-speech output or by a Refreshable Braille display. Student can easily navigate uCertify course using JAWS shortcut keys.

8. State of the Art Educator Tools

uCertify knows the importance of instructors and provide tools to help them do their job effectively. Instructors are able to clone and customize course. Do ability grouping. Create sections. Design grade scale and grade formula. Create and schedule assessments. Educators can also move a student from self-paced to mentor-guided to instructor-led mode in three clicks.

9. Award Winning Learning Platform (LMS)

uCertify has developed an award winning, highly interactive yet simple to use platform. The SIIA CODiE Awards is the only peer-reviewed program to showcase business and education technology's finest products and services. Since 1986, thousands of products, services and solutions have been recognized for achieving excellence. uCertify has won CODiE awards consecutively for last 7 years:

- **2014**

1. Best Postsecondary Learning Solution

- **2015**

1. Best Education Solution
2. Best Virtual Learning Solution
3. Best Student Assessment Solution
4. Best Postsecondary Learning Solution
5. Best Career and Workforce Readiness Solution
6. Best Instructional Solution in Other Curriculum Areas
7. Best Corporate Learning/Workforce Development Solution

- **2016**

1. Best Virtual Learning Solution
2. Best Education Cloud-based Solution
3. Best College and Career Readiness Solution
4. Best Corporate / Workforce Learning Solution
5. Best Postsecondary Learning Content Solution

6. Best Postsecondary LMS or Learning Platform
7. Best Learning Relationship Management Solution

- **2017**

1. Best Overall Education Solution
2. Best Student Assessment Solution
3. Best Corporate/Workforce Learning Solution
4. Best Higher Education LMS or Learning Platform

- **2018**

1. Best Higher Education LMS or Learning Platform
2. Best Instructional Solution in Other Curriculum Areas
3. Best Learning Relationship Management Solution

- **2019**

1. Best Virtual Learning Solution
2. Best Content Authoring Development or Curation Solution
3. Best Higher Education Learning Management Solution (LMS)

- **2020**

1. Best College and Career Readiness Solution
2. Best Cross-Curricular Solution
3. Best Virtual Learning Solution

10. Chapter & Lessons

uCertify brings these textbooks to life. It is full of interactive activities that keeps the learner engaged. uCertify brings all available learning resources for a topic in one place so that the learner can efficiently learn without going to multiple places. Challenge questions are also embedded in the chapters so learners can attempt those while they are learning about that particular topic. This helps them grasp the concepts better because they can go over it again right away which improves learning.

Learners can do Flashcards, Exercises, Quizzes and Labs related to each chapter. At the end of every lesson, uCertify courses guide the learners on the path they should follow.

Syllabus

Chapter 1: Introduction

- Who Should Read This Course?
- Should I Learn C First?
- Why Should I Learn C++?
- What If I Don't Want This Course?
- Conventions Used in This Course

Chapter 2: Hour 1: Writing Your First Program

- Using C++
- Compiling and Linking the Source Code
- Creating Your First Program
- Summary

Chapter 3: Hour 2: Organizing the Parts of a Program

- Reasons to Use C++
- The Parts of a Program
- Comments

- Functions
- Summary

Chapter 4: Hour 3: Creating Variables and Constants

- What Is a Variable?
- Defining a Variable
- Assigning Values to Variables
- Using Type Definitions
- Constants
- Attributes
- Knowledge Check
- Auto-Typed Variables
- Summary

Chapter 5: Hour 4: Using Expressions, Statements, and Operators

- Statements
- Expressions
- Operators

- If-Else Conditional Statements
- Logical Operators
- Tricky Expression Values
- Summary

Chapter 6: Hour 5: Calling Functions

- What Is a Function?
- Declaring and Defining Functions
- Using Variables with Functions
- Function Parameters
- Returning Values from Functions
- Default Function Parameters
- Overloading Functions
- Auto-Typed Return Values
- Summary

Chapter 7: Hour 6: Controlling the Flow of a Program

- Looping
- while Loops

- do-while Loops
- for Loops
- switch Statements
- Summary

Chapter 8: Hour 7: Storing Information in Arrays and Strings

- What Is an Array?
- Writing Past the End of Arrays
- Initializing Arrays
- Multidimensional Arrays
- Character Arrays
- Array bounded/unbounded
- Copying Strings
- Reading Arrays with Foreach Loops
- Summary

Chapter 9: Hour 8: Creating Basic Classes

- What Is a Type?

- Creating New Types
- Classes and Members
- Accessing Class Members
- Private Versus Public Access
- Implementing Member Functions
- Creating and Deleting Objects
- Summary

Chapter 10: Hour 9: Moving into Advanced Classes

- const Member Functions
- Interface Versus Implementation
- Organizing Class Declarations and Function Definitions
- Inline Implementation
- Classes with Other Classes as Member Data
- Summary

Chapter 11: Hour 10: Creating Pointers

- Understanding Pointers and Their Usage
- The Stack and the Heap

- Null Pointer Constant
- Summary

Chapter 12: Hour 11: Developing Advanced Pointers

- Creating Objects on the Heap
- Deleting Objects
- Accessing Data Members Using Pointers
- Member Data on the Heap
- The this Pointer
- Stray or Dangling Pointers
- const Pointers
- const Pointers and const Member Functions
- Summary

Chapter 13: Hour 12: Creating References

- What is a Reference?
- Creating a Reference
- Using the Address of Operator on References

- What Can Be Referenced?
- Null Pointers and Null References
- Passing Function Arguments by Reference
- Understanding Function Headers and Prototypes
- Returning Multiple Values
- Summary

Chapter 14: Hour 13: Developing Advanced References and Pointers

- Passing by Reference for Efficiency
- Passing a const Pointer
- References as an Alternative to Pointers
- When to Use References and When to Use Pointers
- References to Objects Not in Scope
- Returning a Reference to an Object on the Heap
- Pointer, Pointer, Who Has the Pointer?
- Summary

Chapter 15: Hour 14: Calling Advanced Functions

- Overloaded Member Functions

- Using Default Values
- Initializing Objects
- The Copy Constructor
- Compile-Time Constant Expressions
- Summary

Chapter 16: Hour 15: Using Operator Overloading

- Operator Overloading
- Conversion Operators
- Summary

Chapter 17: Hour 16: Extending Classes with Inheritance

- What Is Inheritance?
- Private Versus Protected
- Constructors and Destructors
- Passing Arguments to Base Constructors
- Overriding Functions
- Summary

Chapter 18: Hour 17: Using Polymorphism and Derived Classes

- Polymorphism Implemented with Virtual Member Functions
- How Virtual Member Functions Work
- Summary

Chapter 19: Hour 18: Making Use of Advanced Polymorphism

- Problems with Single Inheritance
- Abstract Data Types
- Summary

Chapter 20: Hour 19: Storing Information in Linked Lists

- Linked Lists and Other Structures
- Linked List Case Study
- Linked Lists as Objects
- Summary

Chapter 21: Hour 20: Using Special Classes, Functions, and Pointers

- Static Member Data
- Static Member Functions

- Containment of Classes
- Friend Classes and Functions
- Summary

Chapter 22: Hour 21: Using New Features of C++23

- The Newest Version of C++
- Using auto in Function Return Types
- Improved Numeric Literals
- The constexpr Keyword
- Ranges Library
- Lambda Expressions
- Coroutines
- Modules
- Summary

Chapter 23: Hour 22: Employing Object-Oriented Analysis and Design

- The Development Cycle
- Simulating an Alarm System

- PostMaster: A Case Study
- Summary

Chapter 24: Hour 23: Creating Templates

- What Are Templates?
- Instances of the Template
- Template Definition
- Using Template Items
- Fold Expressions
- Summary

Chapter 25: Hour 24: Dealing with Exceptions and Error Handling

- Bugs, Errors, Mistakes, and Code Rot
- Handling the Unexpected
- Exceptions
- Using try and catch Blocks
- Writing Professional-Quality Code
- Summary

Chapter 26: Appendix A: Binary and Hexadecimal

- Other Bases
- Around the Bases
- Hexadecimal

Chapter 27: Appendix B: Using the MinGW C++ Compiler on Windows

- Downloading MinGW-w64
- Setting the Path Environment Variable

11. Practice Test

uCertify provides full length practice tests. These tests closely follow the exam objectives and are designed to simulate real exam conditions. Each course has a number of test sets consisting of hundreds of items to ensure that learners are prepared for the certification exam.

Here's what you get

60

PRE-ASSESSMENTS QUESTIONS

60

POST-ASSESSMENTS QUESTIONS

Features

Each question comes with detailed remediation explaining not only why an answer option is correct but also why it is incorrect.

Unlimited Practice

Each test can be taken unlimited number of times until the learner feels they are prepared. Learner can review the test and read detailed remediation. Detailed test history is also available.

Each test set comes with learn, test and review modes. In learn mode, learners will attempt a question and will get immediate feedback and complete remediation as they move on to the next question. In test mode, learners can take a timed test simulating the actual exam conditions. In review mode, learners can read through one item at a time without attempting it.

12. Performance Based Labs

uCertify's performance-based labs are simulators that provides virtual environment. Labs deliver hands on experience with minimal risk and thus replace expensive physical labs. uCertify Labs are cloud-based, device-enabled and can be easily integrated with an LMS. Features of uCertify labs:

- Provide hands-on experience in a safe, online environment
- Labs simulate real world, hardware, software & CLI environment
- Flexible and inexpensive alternative to physical Labs
- Comes with well-organized component library for every task
- Highly interactive - learn by doing
- Explanations and remediation available
- Videos on how to perform

Lab Tasks

- Creating Your First Program
- Understanding Preprocessor Directives

- Using Constants
- Using the If-Else Conditional Statement
- Using the Compound If Statements
- Using the if Statement
- Using Local Variables with Functions
- Using the Function Parameters
- Using Function Overloading
- Understanding the continue Statement
- Using the while Loop
- Using the do-while Loop
- Using the Nested Loops
- Using the switch Statement
- Initializing Arrays
- Using an Array
- Implementing a Character Array
- Storing Data in an Array
- Implementing Member Functions
- Implementing Member Variables and Member Functions
- Creating Objects
- Understanding the const Member Function
- Understanding Pointers
- Using the new and delete Keywords
- Using the Arrow Operator
- Understanding const Pointers and const Member Functions
- Creating a Reference
- Returning by Reference
- Passing a const Pointer
- Understanding the Copy Constructor
- Overloading the Addition Operator
- Overriding a Function
- Hiding the Base Class Member Function
- Implementing Polymorphism Using the Virtual Function
- Using the dynamic_cast Operator
- Understanding Abstract Data Types
- Using Static Member Functions

- Understanding Pointers to Member Functions
- Using auto in Function Return Types
- Using the Lambda Expression
- Using Templates
- Dealing with Exceptions

Here's what you get



13. Post-Assessment

After completion of the uCertify course Post-Assessments are given to students and often used in conjunction with a Pre-Assessment to measure their achievement and the effectiveness of the exam.

GET IN TOUCH:

 3187 Independence Drive
Livermore, CA 94551,
United States



+1-415-763-6300



support@ucertify.com



www.ucertify.com