

# uCertify

## Course Outline

### Beginning Programming All-in-One



17 Jul 2024

1. Course Objective
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: Getting Started Programming a Computer  
Chapter 3: Different Methods for Writing Programs  
Chapter 4: Types of Programming Languages  
Chapter 5: Programming Tools  
Chapter 6: How Programs Work  
Chapter 7: Variables, Data Types, and Constants  
Chapter 8: Manipulating Data  
Chapter 9: Making Decisions by Branching  
Chapter 10: Repeating Commands by Looping  
Chapter 11: Breaking a Large Program into Subprograms  
Chapter 12: Breaking a Large Program into Objects  
Chapter 13: Reading and Saving Files  
Chapter 14: Documenting Your Program  
Chapter 15: Principles of User Interface Design  
Chapter 16: Debugging and Testing  
Chapter 17: Structures and Arrays  
Chapter 18: Sets and Linked Lists  
Chapter 19: Collections and Dictionaries

Chapter 20: Stacks, Queues, and Deques  
Chapter 21: Graphs and Trees  
Chapter 22: Sorting Algorithms  
Chapter 23: Searching Algorithms  
Chapter 24: String Searching  
Chapter 25: Data Compression Algorithms  
Chapter 26: Encryption Algorithms  
Chapter 27: HyperText Markup Language  
Chapter 28: CSS  
Chapter 29: JavaScript  
Chapter 30: PHP  
Chapter 31: Ruby  
Chapter 32: C and C++  
Chapter 33: Java and C#  
Chapter 34: Perl and Python  
Chapter 35: Kotlin  
Chapter 36: Swift and SwiftUI  
Chapter 37: Flutter and Dart  
Chapter 38: Database Management  
Chapter 39: Bioinformatics  
Chapter 40: Computer Security  
Chapter 41: Artificial Intelligence  
Chapter 42: Mobile and Wearable Computing  
Chapter 43: Game Engines  
Chapter 44: The Future of Computer Programming

Videos and How To

8. Live labs

Lab Tasks

Here's what you get

## 1. Course Objective

Beginning Programming All-in-One For Dummies is your gateway to the world of programming. Whether you're a complete beginner or have some prior knowledge, this course offers a comprehensive introduction to computer programming fundamentals and various programming languages. It provides interactive lessons, quizzes, and hands-on labs, ensuring an engaging and practical learning experience. This course equips you with the essential skills to understand the basics of coding and problem-solving, making it easy to dive into the world of computer programming.

## 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.

**72**  
EXERCISES

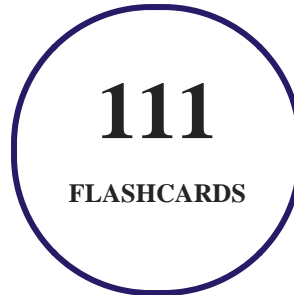
## 3. Quiz

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.

**190**  
QUIZ

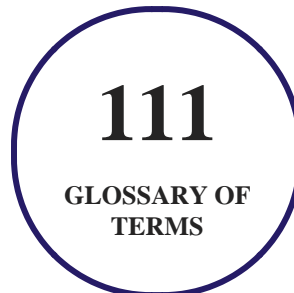
## 4. flashcards

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

- About This Course
- False Assumptions
- Icons Used in This Course
- Where to Go from Here

#### Chapter 2: Getting Started Programming a Computer

- How Computer Programming Works

- The History of Computer Programming
- Figuring Out Programming

### Chapter 3: Different Methods for Writing Programs

- Spaghetti Programming
- Structured Programming
- Event-Driven Programming
- Object-Oriented Programming
- Using Protocol-Oriented Programming
- Design Patterns

### Chapter 4: Types of Programming Languages

- Your First Language
- Curly-Bracket Languages
- Artificial Intelligence Languages
- Scripting Languages
- Database Programming Languages
- Comparing Programming Languages

## Chapter 5: Programming Tools

- Choosing a Compiler
- Finding an Interpreter
- Compiling to a Virtual Machine
- Writing a Program with an Editor
- Fixing a Program with a Debugger
- Saving Time with Third-Party Components
- Optimizing a Program with a Profiler
- Managing Source Code
- Creating a Help File
- Installing a Program
- Dissecting Programs with a Disassembler

## Chapter 6: How Programs Work

- Using Keywords as Building Blocks
- Organizing a Program
- Dividing a Program into Subprograms
- Dividing a Program into Objects

- Creating a User Interface

## Chapter 7: Variables, Data Types, and Constants

- Using Different Data Types
- Storing Data in a Variable
- Retrieving Data from a Variable
- Using Constant Values
- Defining the Scope of a Variable

## Chapter 8: Manipulating Data

- Storing Data with the Assignment Operator
- Using Math to Manipulate Numbers
- Manipulating Strings
- Finding Strings with Regular Expressions
- Using Comparison Operators
- Using Boolean Operators
- Converting Data Types

## Chapter 9: Making Decisions by Branching

- Picking One Choice with the IF-THEN Statement
- Picking Two Choices with the IF-THEN-ELSE Statement
- Picking Three or More Choices with the IF-THEN-ELSEIF Statement
- Playing with Multiple Boolean Operators
- Making Multiple Choices with the SELECT CASE Statement

## Chapter 10: Repeating Commands by Looping

- Looping a Fixed Number of Times with the FOR-NEXT Loop
- Looping Zero or More Times with the WHILE Loop
- Looping at Least Once with the DO Loop
- Playing with Nested Loops
- Prematurely Exiting from a Loop
- Checking Your Loops

## Chapter 11: Breaking a Large Program into Subprograms

- Creating and Using Subprograms
- Passing Parameters
- Repeating a Subprogram with Recursion

## Chapter 12: Breaking a Large Program into Objects

- How Object-Oriented Programming Works
- Encapsulation Isolates Data and Subprograms
- Sharing Code with Inheritance
- Polymorphism: Modifying Code without Changing Its Name
- Design Patterns
- Object-Oriented Languages
- Real-Life Programming Examples

## Chapter 13: Reading and Saving Files

- Storing Data in Text Files
- Storing Fixed-Size Data in Random-Access Files
- Storing Varying-Size Data in Untyped Files
- Using Database Files

## Chapter 14: Documenting Your Program

- Adding Comments to Source Code
- Writing Software Documentation

## Chapter 15: Principles of User Interface Design

- The Evolution of User Interfaces
- Elements of a User Interface
- Designing a User Interface

## Chapter 16: Debugging and Testing

- Common Types of Programming Errors
- Debugging with Comments and Print Statements
- Breakpoints, Stepping, and Watching
- Testing Code

## Chapter 17: Structures and Arrays

- Using Structures
- Using an Array
- Working with Resizable Arrays
- Working with Multidimensional Arrays
- Using Structures with Arrays
- Drawbacks of Arrays

## Chapter 18: Sets and Linked Lists

- Using Sets
- Using Linked Lists
- Drawbacks of Sets and Linked Lists

## Chapter 19: Collections and Dictionaries

- Using a Collection
- Using Dictionaries
- Understanding Hash Tables

## Chapter 20: Stacks, Queues, and Deques

- Using Stacks
- Using Queues
- Using Deques

## Chapter 21: Graphs and Trees

- Understanding Graphs
- Creating Trees
- Taking Action on Trees



## Chapter 22: Sorting Algorithms

- Using Bubble Sort
- Using Selection Sort
- Using Insertion Sort
- Using Shell Sort
- Using Heap Sort
- Using Merge Sort
- Using Quick Sort
- Comparing Sorting Algorithms

## Chapter 23: Searching Algorithms

- Sequential Search
- Using Indexes
- Adversarial Search

## Chapter 24: String Searching

- Sequential Text Search
- Searching with Regular Expressions

- Searching Phonetically

## Chapter 25: Data Compression Algorithms

- Lossless Data Compression Algorithms
- Lossy Data Compression

## Chapter 26: Encryption Algorithms

- How Encryption Works
- The Basics of Encryption
- Symmetric/Asymmetric Encryption Algorithms
- Cracking Encryption

## Chapter 27: HyperText Markup Language

- The Structure of an HTML Document
- Adding Graphics
- Defining the Background
- Creating Hyperlinks
- Making Tables

## Chapter 28: CSS

- The Structure of a Stylesheet
- Creating Style Classes
- Separating Styles in Files
- Cascading Stylesheets

## Chapter 29: JavaScript

- The Structure of a JavaScript Program
- Creating Comments
- Declaring Variables
- Using Operators
- Branching Statements
- Looping Statements
- Creating Functions
- Using Arrays
- Designing User Interfaces

## Chapter 30: PHP

- Examining the Structure of a PHP Program
- Creating Comments
- Declaring Variables
- Using Operators
- Branching Statements
- Looping Statements
- Creating Functions
- Using Arrays
- Creating Objects

## Chapter 31: Ruby

- The Structure of a Ruby Program
- Creating Comments
- Declaring Variables
- Using Operators
- Branching Statements
- Looping Statements
- Creating Functions

- Using Data Structures
- Creating Objects

## Chapter 32: C and C++

- Looking at the Structure of a C/C++ Program
- Creating Comments
- Declaring Variables
- Using Operators
- Branching Statements
- Looping Statements
- Creating Functions
- Data Structures
- Using Objects

## Chapter 33: Java and C#

- Looking at the Structure of a Java/C# Program
- Creating Comments
- Declaring Variables
- Using Operators

- Branching Statements
- Looping Statements
- Creating Functions
- Data Structures
- Using Objects

## Chapter 34: Perl and Python

- Reviewing the Structure of a Perl or Python Program
- Creating Comments
- Defining Variables
- Using Operators
- Branching Statements
- Looping Statements
- Creating Functions
- Making Data Structures
- Using Objects

## Chapter 35: Kotlin

- Looking at the Structure of a Kotlin Program
- Creating Comments
- Declaring Variables
- Declaring Constants
- Using Operators
- Branching Statements
- Looping Statements
- Creating Functions
- Creating Data Structures
- Creating Objects

## Chapter 36: Swift and SwiftUI

- Considering the Structure of a Swift Program
- Creating Comments
- Declaring Variables
- Declaring Constants
- Using Operators
- Branching Statements

- Looping Statements
- Creating Functions
- Data Structures
- Creating Objects

## Chapter 37: Flutter and Dart

- Working with Flutter
- Understanding the Dart Language

## Chapter 38: Database Management

- Understanding the Basics of Databases
- Manipulating Data
- Database Programming

## Chapter 39: Bioinformatics

- The Basics of Bioinformatics
- Database Searches
- Bioinformatics Programming

## Chapter 40: Computer Security



- Stopping Malware
- Stopping Hackers
- Secure Computing

## Chapter 41: Artificial Intelligence

- Problem Solving
- Machine Learning
- Applications of Artificial Intelligence

## Chapter 42: Mobile and Wearable Computing

- Understanding the Different Generations of Computing
- Giving Data to the User
- Getting Data from the User
- Tracking Motion and Location
- Tracking Real-Time Health Data
- Looking to the Future of Augmented Reality and Wearable Computers

## Chapter 43: Game Engines

- Understanding Game Engines

- Picking a Game Engine
- Programming a Game Engine
- Exploring the Future Uses of Game Engines

## Chapter 44: The Future of Computer Programming

- Picking a Programming Language
- Picking an Operating System
- Doing Cross-Platform Programming
- Defining Your Future in Programming

## 11. Live Labs

The benefits of live-labs are:

- Exam based practical tasks
- Real equipment, absolutely no simulations
- Access to the latest industry technologies
- Available anytime, anywhere on any device
- Break and Reset functionality
- No hardware costs

## Lab Tasks

## **Getting Started Programming a Computer**

- Displaying a Sample Message

## **Different Methods for Writing Programs**

- Creating a Textbox and a Progress Bar
- Creating an Object

## **Variables, Data Types, and Constants**

- Storing Data
- Manipulating the Variable
- Concatenating Strings in Visual Basic

## **Manipulating Data**

- Computing Leaves of an Employee
- Understanding Arithmetic Operations
- Manipulating Strings
- Using Comparison Operators

## **Making Decisions by Branching**

- Using the if Statement
- Using the IF ELSE Statement
- Using the IF-THEN-ELSEIF Statement
- Using the If-Elif-Else Statement

## **Repeating Commands by Looping**

- Using the for Loop
- Using the while Loop
- Using the do-while Loop
- Using a Nested Loop

## **Breaking a Large Program into Subprograms**

- Using Python Functions
- Passing Parameters to a Java Program
- Finding the Factorial of a Number Using Recursion

### **Breaking a Large Program into Objects**

- Overloading a Method

### **Reading and Saving Files**

- Reading Data from a File

### **Debugging and Testing**

- Debugging a Sample Python Code

### **Structures and Arrays**

- Declaring an Array
- Declaring a Multidimensional Array

### **Sets and Linked Lists**

- Using Sets
- Creating a Linked List

### **Collections and Dictionaries**

- Using Operators in Visual Basic
- Adding Data to a Dictionary

### **Stacks, Queues, and Deques**

- Using Stacks
- Creating a Priority Queue

### **Sorting Algorithms**

- Implementing Bubble Sort

- Implementing Selection Sort
- Implementing Insertion Sort
- Implementing Heap Sort
- Implementing Merge Sort
- Implementing Quick Sort

### **Searching Algorithms**

- Performing Binary Search
- Performing Sequential Search

### **String Searching**

- Using Regular Expressions to Replace Text

### **Encryption Algorithms**

- Using Symmetric Algorithm
- Using an Asymmetric Algorithm

### **HyperText Markup Language**

- Using HTML
- Using the Font Color in HTML

### **CSS**

- Using HTML and CSS
- Modifying Elements Using CSS Margins, Borders, and Padding
- Positioning Images Relative to Text Using CSS

### **JavaScript**

- Using the Arithmetic Operators
- Using the for Loop

### **PHP**

- Using PHP

## **Ruby**

- Creating Objects in Ruby
- Determining a Number as Even or Odd

## **C and C++**

- Using Variables
- Using Operators

## **Java and C#**

- Using the int Type
- Creating Functions in Java

## **Perl and Python**

- Determining if a Number is Prime or Not

## **Kotlin**

- Using the Kotlin Language

## **Flutter and Dart**

- Creating a Function in Dart

## **Database Management**

- Using the SELECT Statement
- Using the DELETE Statement
- Using the UPDATE Statement

## **Computer Security**

- Scanning and Classifying Different Types of Viruses
- Analyzing Malware

## Here's what you get

**65**

LIVE LABS

**11**

VIDEO TUTORIALS

**24**

MINUTES

**GET IN TOUCH:**



3187 Independence Drive  
Livermore, CA 94551,  
United States



+1-415-763-6300



support@ucertify.com



www.ucertify.com