

Unity Game Programming

Course Syllabus and Planner

Course Overview

The CompuScholar **Unity Game Programming** curriculum is a one-year (two-semester) course covering topics typically found in **Video Game Design** or similar classes. This course has been aligned to specific course standards in many states. Please visit our course description page for a video tour and alignment information.

<https://www.compuscholar.com/schools/courses/unity/>

Other introductory programming courses are not required; students merely need to have typical computer usage skills prior to starting this course.

Course Material

The course consists of the following student-facing elements:

- **Instructional Videos** – optional (not required), but enjoyed by many students as an audio-visual introduction to the lesson topics.
- **Lesson Text** – required reading, contains full topic details and coding exercises
- **Quizzes and Exams** – multiple-choice and automatically graded by our system
- **Chapter Activities** – hands-on projects, submitted for a grade

Teachers additionally have access to:

- **Teacher's Guides** – for each lesson, with suggested discussion questions
- **Quiz and Exam Answer Keys** – PDFs for quick reference
- **Activity Solution Guides** – fully coded activity solutions for each chapter activity

Labs and Programming Environment

Every chapter contains one or more hands-on programming labs where students will design or implement programs to demonstrate understanding of the lesson topics. Students will get the opportunity to work on individual and group projects and will experience all phases of a project lifecycle, including requirements, design, implementation, and testing.

The chosen gaming framework is Unity (<https://unity3d.com/>). **Schools or students are required to install and license the Unity IDE and associated scripting editor (Visual Studio) on a compatible Windows or MacOS computer.** The course contains detailed download, installation, and licensing instructions for the Unity IDE with a FREE personal or educational license.

Course Planner

A typical school year consists of approximately 36 calendar weeks or 180 days of school. The course plan covers approximately 170 school days, with additional time allocated for review, make-up work, or individual projects. Each “day” listed below represents one typical class period of 45 – 60 minutes, so students will typically work 3-5 hours per week. Some classes may move faster or slower than the suggested pace.

Each chapter contains multiple lesson quizzes and a chapter test in addition to the listed Lab assignments. Teachers may choose to add Supplemental Lessons as desired to meet state standards or student interest. Mid-term and final project timelines are flexible and may be scaled “up” or “down” to match the available class-time.

Days	Reading and Objectives	Labs
5	Chapter One: Game Engines <ul style="list-style-type: none"> • Engine Concepts • Development Tools • Introducing Unity 	Install Unity Software

Days	Reading and Objectives	Labs
5	Chapter Two: Unity Development Environment <ul style="list-style-type: none"> • IDE Basics • Unity Concepts • Sprites 	Flock of Birds
7	Chapter Three: Introduction to Scripting <ul style="list-style-type: none"> • C# Language Concepts • Creating Scripts • C# Coding Fundamentals • Game Loops and Functions 	Reporting for Duty
5	Chapter Four: Simple Movement and Input <ul style="list-style-type: none"> • Simple Movement • Simple Rotation and Scaling • Easy Input Handling in Unity 	Alien Dance Squad
7	Chapter Five: 2D Physics Concepts <ul style="list-style-type: none"> • Rigidbody Components • Unity Colliders • Physics Materials • Scripting Collision Events 	Physics Pinball
6	Chapter Six: Primitive Data and Math <ul style="list-style-type: none"> • Data Types and Variables • Mathematical Operations • Variable Scope and Access • Displaying Data 	Pinball Scoring

Days	Reading and Objectives	Labs
5	Chapter Seven: Decisions and Flow Control <ul style="list-style-type: none"> • Logical Expressions • "if/else" Statements • "switch" Statements 	Thunder Road
7	Chapter Eight: Organizing Game Objects <ul style="list-style-type: none"> • Parent-Child Objects • Sorting Layers • Tagging Game Objects • Collision Layers 	Mower Dodgeball
10	Chapter Nine: Object-Oriented Concepts <ul style="list-style-type: none"> • Defining Classes • Creating and Using Classes • Defining Functions • Accessing Game Objects • Constructor and Property Functions 	Deep Space
7	Chapter Ten: Managing Game Objects <ul style="list-style-type: none"> • Prefabs • Creating and Destroying Objects • Activating and Deactivating Objects • Controlling Object Lifespans with Invoke 	Deep Space 2

Days	Reading and Objectives	Labs
5	Chapter Eleven: Exceptions and Debugging <ul style="list-style-type: none"> • Run-Time Exceptions • Finding Run-time Errors • Using the Debugger 	Bug Hunt
5	Chapter Twelve: Loops and Arrays <ul style="list-style-type: none"> • Arrays • for() and foreach() Loops • while() Loops 	Banana Breakout
6	Chapter Thirteen: Game Design Strategies <ul style="list-style-type: none"> • Game Requirements • Game Mechanics • Storytelling and Progression • Design Documents 	Planning Documents
10	Chapter Fourteen: Mid-Term Project <ul style="list-style-type: none"> • Kickoff 	Mid-Term Requirements Mid-Term Design Mid-Term Coding & Testing
7	Chapter Fifteen: Virtual Worlds <ul style="list-style-type: none"> • Moving Cameras • Setting Boundaries • Building a Tile World • Mini-Maps 	Treasure Hunt

Days	Reading and Objectives	Labs
6	Chapter Sixteen: Scrolling Games <ul style="list-style-type: none"> • Wrapping Background • Scrolling Game Mechanics • Parallax Effects 	RoboDash
7	Chapter Seventeen: Animation <ul style="list-style-type: none"> • Simple Unity Animation • Animator States • Scripting Animations • Animations and Colliders 	RoboDash Animation
6	Chapter Eighteen: Sound Effects <ul style="list-style-type: none"> • Sound Files • Adding Sounds to Game Objects • Scripting Sounds 	RoboDash Sounds
6	Chapter Nineteen: Advanced Game Physics <ul style="list-style-type: none"> • Applying Forces • Unity Physics Joints • Unity 2D Effectors 	Mini-Golf
6	Chapter Twenty: Multiple Scenes <ul style="list-style-type: none"> • Creating New Scenes • Scripting Scene Changes • Saving Objects Across Scenes 	Mini-Golf Levels

Days	Reading and Objectives	Labs
6	Chapter Twenty-One: Artificial Intelligence <ul style="list-style-type: none"> • Artificial Intelligence Concepts • Flowcharts and Algorithms • Scripting AI 	Space Creeps
6	Chapter Twenty-Two: User Interfaces <ul style="list-style-type: none"> • Unity Buttons • Other UI Controls • UI Design Concepts 	Space Creeps Settings
5	Chapter Twenty-Three: Game Art <ul style="list-style-type: none"> • Perspectives • Color Theory • Image Editing 	Image Replacements
6	Chapter Twenty-Four: Publishing Games <ul style="list-style-type: none"> • Splash Screens, Credit Scenes and Icons • Publishing to PC, Mac and Linux Computers • Publishing to Smartphones • Publishing to Game Consoles 	Publish Your Game

Days	Reading and Objectives	Labs
5	Chapter Twenty-Five: Software Development Lifecycles and Teamwork <ul style="list-style-type: none"> • Software Lifecycles • Internal and External Documentation • Software Teams and Tools 	Project Planning
10	Chapter Twenty-Six: Final Project <ul style="list-style-type: none"> • Kickoff 	Final Requirements Final Design Final Coding & Testing

Supplemental Chapters and Lessons are described on the next page.

Supplemental Chapters

Supplemental Chapters and lessons can be used as desired to meet state standards or provide enrichment topics for students. Lessons are loosely organized into chapters but can be skipped or completed in any order.

Days	Reading and Objectives	Labs
3	Supplemental Chapter One: Ethics and Society <ul style="list-style-type: none"> • Computing Ethics • Intellectual Property • Security 	N/A
2	Supplemental Chapter Two: Video Game History <ul style="list-style-type: none"> • Types of Games • The Evolution of Game Consoles 	N/A
13	Supplemental Chapter Three: Additional Topics <ul style="list-style-type: none"> • Collaboration and PIM Tools • The Binary Number System • ESRB Ratings • Beyond Gaming • Gaming and Software Careers • Video Displays • Student Organizations • Advanced Game Research 	Productivity Report Science Project Exploring Gaming Careers CTSO Exploration Game Analysis