

CompuScholar, Inc.
Alignment to
"Unity Certified User: Programmer"
Certification Exam Requirements

Unity Exam Details:

Exam Title:	Unity Certified User: Programmer
Exam Link:	https://unity.com/products/unity-certifications/user-programmer

CompuScholar Course Details:

Course Title:	Unity Game Programming
Course ISBN:	978-0-9887070-7-8
Course Year:	2022

Note 1: Citation(s) listed may represent a subset of the instances where objectives are met throughout the course.

Note 2: Citation(s) for a "Lesson" refer to the "Lesson Text" elements and associated "Activities" within the course, unless otherwise noted. The "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts, and the Lesson Text contains full details.

Course Description

"This certification is designed for future developers who want to create their own game or apps, or plan to attend a postsecondary program in game development, STEM, or animation." (from the Exam Link)

Exam Requirements

1. Debugging, problem-solving, and interpreting the API	CITATION(S)
1.1. Given an example of a debug log message, create the code that created the log message.	Chapter 3, Lesson 3
1.2. Given a code clip and its associated error message(s), determine which object(s) is(are) null.	Chapter 11, Lessons 1, 2
1.3. Given a specific programming task requiring the use of a particular class in the API, determine the appropriate method and/or properties, arguments, or other syntax to use.	Throughout the course

2. Creating code	CITATION(S)
2.1. Indicate when and how to initialize and use variables including but not limited to the appropriate use of all variable modifiers and data collections such as Arrays, Lists and Dictionaries.	Chapter 6 (Variables) Chapter 12 (Arrays) N/A - Lists/Dictionaries
2.2. Given a list of keywords and syntax elements, construct a viable Function declaration.	Chapter 9, Lesson 3
2.3. Given a code clip and a description of its desired result, identify the appropriate function to control or trigger a state including but not limited to the Animator Controller.	Chapter 17, Lesson 3
2.4. Given a scenario where a specific type of input is required and the building blocks needed are provided, construct the necessary input listener including but not limited to the keyboard and touch input.	Chapter 4, Lesson 3
2.5. Demonstrate when and/or how to use the various logic and flow control operators used in C# and Unity.	Chapter 7
2.6. Given a scenario, identify appropriate actions to take when a UI element reports a change.	Chapter 22, Lessons 1, 2

3. Evaluating Code	CITATION(S)
3.1. Given a scenario about the need to manage an event function, determine the appropriate action to take including but not limited to the keyboard and touch input.	Chapter 3, Lesson 4 Chapter 4, Lesson 3
3.2. Given a code clip that produces an error because of a variable whose data type is declared incorrectly, identify the error.	Chapter 6, Lessons 1, 2 Chapter 11, Lesson 2
3.3. Given a code clip that produces an error because a function or variable is declared or used incorrectly (public/private mismatch), identify the error including but not limited to the use of Animation events.	Chapter 9
3.4. Given a code clip containing a class definition, distinguish whether the class is an ECS class or some other type of class.	N/A
3.5. Given a set of code clips, recognize the clip that uses naming conventions that observe Unity naming standards.	Camel/Pascal case demonstrated but not called out as a Unity standard - see Chapter 6, Lesson 1 Chapter 9, Lesson 1
3.6. Given a code clip (or a set of code clips), recognize the comments that accurately describe what the code is doing.	Chapter 3, Lesson 3

4. Navigating the Interface	CITATION(S)
4.1. Describe the purpose, features, and functions of the various Unity IDE windows.	Chapter 2
4.2. Demonstrate how to change the default scripting IDE.	Chapter 3, Lesson 2

4.3. Given a scenario which includes the following, then create a functional state machine. a. a limited portion of a gaming scenario b. a set of animation clips c. a list of property settings	Chapter 17
4.4. Create and program a function state machine within the Unity Animator Controller including but not limited to the use of Animator functions syntax.	Chapter 17, Lesson 3