
[COURSE NUMBER] COURSE TITLE
WINTER 2022

SECTION:	00001 & 00002	TEACHER:	Instructor Name
PONDERATION:	2-4-3 (Theory-Lab-Homework)	OFFICE:	Office number
PREREQUISITES:	All program courses semesters 1-4	PHONE:	514.999.9999
		E-MAIL:	username@mycollege.ca
		OFFICE HOURS:	By appointment

COURSE DESCRIPTION

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

KEY LEARNING OUTCOMES

- This is a first objective
- This is a second objective
- This is a third objective
- And forth
- And fifth

COURSE-LEVEL LEARNING OUTCOME

The following is a list of course-level learning outcomes.

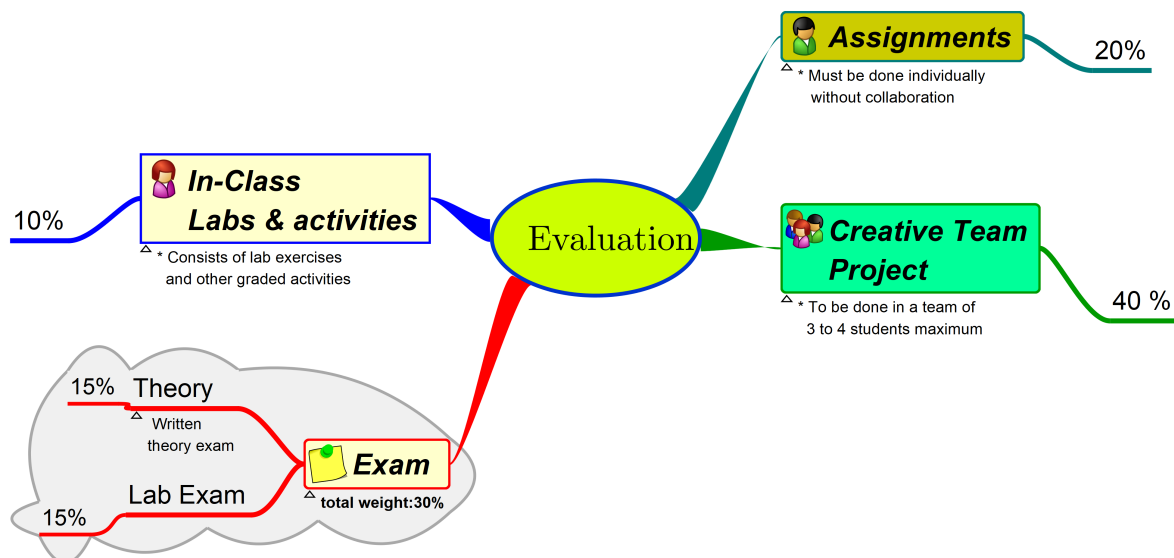
- * Outcome 1
- * Outcome 2
- * Outcome 3
- * Outcome 4
- * Outcome 5

REQUIRED BACKGROUND AND PREREQUISITE KNOWLEDGE

The practical programming components of this course require the use of the C# programming language and .NET Framework. Therefore, students are required to have a reasonably solid knowledge of C#/.NET. Much of the material covered in Programming I & II as well as Programming Patterns is directly relevant for this course, therefore, it is critical that students understand and remember the knowledge they acquired in previous courses. Some of the essential concepts that students should know well to do well in this course include, but not limited, to the following:

- | Core principles of Object-Oriented Programming (OOP)
- | Event-driven programming model: generating and handling events in a GUI context
- | Common data structures such as list, array, hash map, stack, queue, etc.
- | Version-control system and management of changes to files in a collaborative environment

GRADING SCHEME



Note: For a tabular format of this grading scheme, refer to Section 10 that is listed in the Appendix below.


NOTES AND CONSTRAINTS

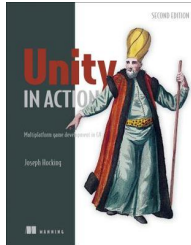
- 📌 In order to pass the course, students must...
- 📌 Late work is not accepted unless prior arrangement with the teacher...
- 📌 Assignments are demonstrated in-class as part of the marking process to provide feedback before exams.
- 📌 Students will be informed of the exact date of each test at least one week in advance.

TEXTBOOKS AND COURSE MATERIALS

 **ONLINE RESOURCES:** Lecture notes, APIs documentation and online Web tutorials will be provided to students throughout the semester.

Using Omnivox: Throughout the semester, students will use Omnivox to download material, upload deliverable files, and obtain all course information.

 **TEXTBOOK:** There is no required textbook for this course. However, the following is a list of textbooks that are strongly recommended for this course:



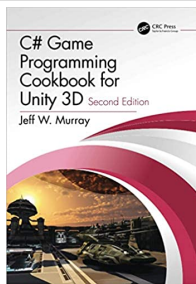
Title: Unity in Action: Multiplatform game development in C#, 2nd Edition

Author: Joseph Hocking

Publisher: Manning Publications, publication year: 2018

ISBN-13: 978-1617294969

Available at: [Manning Publications](#), [Amazon](#) and other vendors.



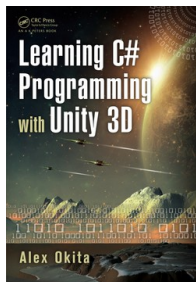
Title: C# Game Programming Cookbook for Unity 3D, 2nd Edition

Author: Jeff W. Murray

Publisher: CRC Press; publication year: 2021

ISBN-13: 978-0367321642

Available at: [Amazon](#), and other vendors.



Title: Learning C# Programming with Unity 3D, 1st Edition

Author: Alex Okita



Publisher: A K Peters/CRC Press; publication year: 2014


ISBN-13: 978-1498760201


Available at: [Amazon](#), and other vendors.

REQUIRED SOFTWARE AND TOOLS


 **Operating system:**  Windows 10,  Linux, **or**  macOS

 **IDE & Game Engine:**  Unity **2020.3 (LTS)** & Visual Studio **2019** (Community Edition)


 **Web Browser:** Google Chrome.


 Markdown for writing documentation.

 Distributed version control system.

 Bitbucket: a web-based version control repository hosting service.

 Trello: a Web-based project management system.

 A storage medium (a USB flash memory or any online free storage service such as GDrive or OneDrive) for storing and backing up your files.

 If you want to work at home on your assignments, students must install and use the **same versions** of the required software that are installed in our labs.

COURSE POLICIES

CLASS POLICY AND RULES OF CONDUCT

- 📄 Laptops are strictly prohibited in classroom during the lectures and labs.
- 📱 **Cell phones strictly prohibited during classes and labs.** Electronic devices must be turned off (🔇) and placed in your bags (not on the desk just in front of you).
- 🎧 **Headphones** should be removed all the time.
- 📝 **During lectures**, students must turn off their monitors and take notes. Using the computer during lectures without authorization is strictly not allowed.
- 📹 Video or audio recordings and taking photographs are NOT permitted.
- 🗣️ Persistent talking, whispering or any disruptive attitude will not be tolerated.

APPENDIX

EVALUATION PROCEDURES

Course Component	Overall Weight	Tentative Period
Individual Homework		
<i>Assignment</i>	20%	Week 2-7
<i>In-class labs</i>	10%	Week 1-11
Exam		
<i>Theoretical exam</i>	15%	Week 7
<i>Lab exam</i>	15%	Week 8
LIA		
<i>Team project</i>	40%	Week 2 to 15

TENTATIVE COURSE CONTENT

Week	Topics & Homework
1	Course outline discussion & team Project Guidelines Lab: activities covering discussed topics.
2	Topics... Lab: activities covering discussed topics. Announcement:... Assignment #1
3	Topics... Lab: activities covering discussed topics. Team project proposal due.
4	Topics... Lab: activities covering discussed topics. Game Design Document due.
5	Topics... Lab: activities covering discussed topics. Team project proposal presentations.
6	Topics...
7	Review for exam. Assignment due.
8	Exam & work on team project.
9	Topics...
10	Topics... Build #1 due.
11	Topics... TCR Checklist due.
12	Topics... Build #2 due.
13	Work on team project.
14	Playtesting session. Build #3 due.
15	Final project demonstrations. Peer review due. QA testing report & user guide due.

NOTE: The above actual dates may be modified due to the requirements of the class. Also, the indicated dates may be moved backward or forward depending on class progress. **Exact dates and instructions will be announced on course webpage.**