COURSE LEARNING OUTCOMES:

COURSE NUMBER & NAME: GDP 1XX Introduction to Game Programming

LECTURE/LAB HOURS: 3 lecture hours per week

CREDITS: 3

PREREQUISITES: GDP 101, CST 115 or CST 161

COURSE DESCRIPTION: This course builds on the fundamental principles of game design and teaches students the basics of game programming using a modern professional game engine. Students will learn the concepts and features of object-oriented programming using a computer language commonly used in the game industry. The course focuses on various techniques used in developing prototypes of 2D and 3D games.

Upon successful completion of this course, students will be able to:

1. Discuss how a modern professional game engine works
2. Understand the syntax of a computer language commonly used in the game industry
3. Understand how to compile and debug programs in a professional integrated development environment (IDE)
4. Discuss various techniques for scripting gameplay
5. Discuss 2D and 3D coordinate space, physics, and collision detection
6. Understand the role of 2D textures, 3D models, audio assets, particle systems, graphical user interface, input systems, and the game camera
7. Locate, discern, and effectively use information to solve issues and/or problems in game design and development

COURSE MATERIALS:

**Textbook** – Unity in Action: Multiplatform game development in C#, 2nd edition, Joe Hocking, Manning Publications, 2018

COURSE REQUIREMENTS:

* The successful completion of quizzes and final exam.
* The completion of in-class assignments and projects.
* The completion of assigned readings and homework.
* Attendance and class participation. Students are expected to attend all classes.

EVALUATION METHODS:

Quizzes 30%

Final Exam 20%

Assignments 50%

Grading System

A 90 or above C 70 - 76

B+ 87 – 89 D+ 67 - 69

B 80 – 86 D 60 - 66

C+ 77 – 79 F below 60

CLASS SCHEDULE**:**

|  |  |  |
| --- | --- | --- |
| Week | Unit/Content | Learning Activities |
| 1 | Chapter 1: Getting to know Unity | Lecture  Class discussion  Homework |
| 2 | Chapter 2: Building a demo that puts you in 3D space | Lecture  Class discussion  Homework |
| 3 | Chapter 3: Adding enemies and projectiles to the 3D game | Lecture  Class discussion  Homework |
| 4 | Chapter 4: Developing graphics for your game | Lecture  Class discussion  Homework |
| 5 | Chapter 5: Building a Memory game using Unity’s 2D functionality | Lecture  Class discussion  Homework |
| 6 | Chapter 6: Creating a basic 2D platformer | Lecture  Class discussion  Homework |
| 7 | Chapter 7: Putting a GUI onto a game | Lecture  Class discussion  Homework |
| 8 | Chapter 8: Creating a third-person 3D game: player movement and animation | Lecture  Class discussion  Homework |
| 9 | Chapter 9: Adding interactive devices and items within the game | Lecture  Class discussion  Homework |
| 10 | Chapter 10: Connecting your game to the internet | Lecture  Class discussion  Homework |
| 11 | Chapter 11: Playing audio: sound effects and music | Lecture  Class discussion  Homework |
| 12 | Chapter 12: Putting the parts together into a complete game | Lecture  Class discussion  Homework |
| 13 | Chapter 13: Deploying your game to players’ devices | Lecture  Class discussion  Homework |
| 14 | Exam Review | Lecture  Class discussion  Homework |
| 15 | Final Exam |  |

SUGGESTED TEACHING METHODOLOGIES:

* Lecture, group discussion, presentations, multimedia/technology, projects, demonstrations, etc.

**CORRELATION OF PROGRAM and GENERAL EDUCATION OUTCOMES, STUDENT OUTCOMES, AND ASSESSMENT METHODS**

|  |  |  |
| --- | --- | --- |
| **Program and General Education Learning Outcomes** | **Course Learning Outcomes** | **Assessment Methods** |
| Define game design and development terminology and processes | * Discuss how a modern professional game engine works * Discuss various techniques for scripting gameplay * Discuss 2D and 3D coordinate space, physics, and collision detection | Written: Exams, assignments.  Verbal: Class discussion and responses, group discussion and responses. |
| Use a modern professional game engine to develop game prototypes | * Discuss how a modern professional game engine works * Understand the syntax of a computer language commonly used in the game industry * Understand how to compile and debug programs in a professional integrated development environment (IDE) * Discuss various techniques for scripting gameplay * Discuss 2D and 3D coordinate space, physics, and collision detection * Understand the role of 2D textures, 3D models, audio assets, particle systems, graphical user interface, input systems, and the game camera | Written: Exams, assignments.  Verbal: Class discussion and responses, group discussion and responses. |
| Communicate effectively in writing, verbal and electronic formats. | * Locate, discern, and effectively use information to solve issues and/or problems in game design and development | Written: Exams, assignments.  Verbal: Class discussion and responses, group discussion and responses. |

REVISED: 1/4/2019