

LANEY COLLEGE COURSE OUTLINE

COLLEGE:		STATE APPROVAL DATE:	05/11/2018
ORIGINATOR:	Kim Bridges	STATE CONTROL NUMBER:	CCC00059 3387
		BOARD OF TRUSTEES APPROVAL DATE:	05/08/2018
		CURRICULUM COMMITTEE APPROVAL DATE:	03/16/2018
		CURRENT EFFECTIVE DATE:	08/01/2018

DIVISION/DEPARTMENT:

1. REQUESTED CREDIT CLASSIFICATION:

Credit - Degree Applicable
Course is not a basic skills course.
Program Applicable

2. DEPT/COURSE NO:

CIS 044

3. COURSE TITLE:

C# for Immersive Design

4. COURSE: Laney New Course

TOP NO. 0707.10*

5. UNITS: 4.000

HRS/WK LEC: 3.00 Total: 52.50

HRS/WK LAB: 3.00 Total: 52.50

HRS/WK TBA:

6. NO. OF TIMES OFFERED AS SELETED TOPIC: AVERAGE ENROLLMENT:

7. JUSTIFICATION FOR COURSE:

This course covers C# which is an a necessary skill for working in immersive design and emerging technology such as AR/VR.

8. COURSE/CATALOG DESCRIPTION

Beginning C# programming for immersive design: Basic unified modeling language (UML) notation in object-oriented software design and development using the C# programming language in the context of the Unity 3D Engine, focusing on the program structure, syntax, constructs, and keywords.

9. OTHER CATALOG INFORMATION

- a. Modular: No If yes, how many modules:
- b. Open entry/open exit: No
- c. Grading Policy: Letter Grade Only
- d. Eligible for credit by Exam: No
- e. Repeatable according to state guidelines: No
- f. Required for degree/certificate (specify):
AR/VR: Immersive Design
- g. Meets GE/Transfer requirements (specify):
- h. C-ID Number: Expiration Date:

- i. Are there prerequisites/corequisites/recommended preparation for this course? Yes
Date of last prereq/coreq validation: 03/16/2018

10. LIST STUDENT PERFORMANCE OBJECTIVES (EXIT SKILLS): (Objectives must define the exit skills required of students and include criteria identified in Items 12, 14, and 15 - critical thinking, essay writing, problem solving, written/verbal communications, computational skills, working with others, workplace needs, SCANS competencies, all aspects of the industry, etc.)(See SCANS/All Aspects of Industry Worksheet.)

Students will be able to:

1. Implement core concepts of C# .NET
2. Explain object-oriented programming
3. Implement content development utilizing C#
4. Implement core methodologies of interactive storytelling
5. Implement C# code to support 3D or interactive aesthetics
6. Support an interactive prototype with C# scripting and programming
7. Explain software technologies including VR and AR platforms

11A. COURSE CONTENT: List major topics to be covered. This section must be more than listing chapter headings from a textbook. Outline the course content, including essential topics, major subdivisions, and supporting details. It should include enough information so that a faculty member from any institution will have a clear understanding of the material taught in the course and the approximate length of time devoted to each. There should be congruence among the catalog description, lecture and/or lab content, student performance objectives, and the student learning outcomes. List percent of time spent on each topic; ensure percentages total 100%.

LECTURE CONTENT:

LECTURE CONTENT:

List Percents

1. Object-oriented programming and C# language syntax for analyzing and modeling business problems
15%
2. Concepts of C# arithmetic operators and control structures: How to analyze and construct algorithms and translate to appropriate control structures of sequence, selection and iteration to design, code, compile, test and debug a program solution in C# 10%
3. Definition and use of data types and variables, arrays & classes to define and use each of them in C# programs 10%
4. Definition and use of control structures and data types in functions for constructing algorithms to solve business problems 15%
5. Proper program syntax and troubleshooting methods and tools to isolate and fix bugs 15%
6. Tools and techniques for building Windows applications with C#; the effective use of software development tools including editors, libraries, and compilers in the C# environment 10%
7. Object-oriented (O/O) programming concepts and how they are used to create object-oriented applications solving business problems 10%
8. Translating data types and variables, arrays & classes to input, output, and files to create object-oriented applications solving business problems 15%

11B. LAB CONTENT:

LAB CONTENT:

List Percents

Demos, assignments, group exercises and projects that reinforce lectures on:

1. Analyze and model business problems and design, code, compile, test and debug program solutions in C# 10%
2. Design and build User Interfaces to support programs; Develop User Interface (UI) or program interaction to meet user requirements 10%
3. Demonstrate the ability to recognize proper program syntax; use a wide range of troubleshooting methods and tools to isolate and fix bugs 15%
4. Build, list, search and modify Arrays 10%
5. Design, code, test and evaluate functions 15%
6. Effectively use software development tools including editors, libraries, and compilers 15%
7. Build Windows applications with C# 15%
8. Demonstrate an ability to use object-oriented (O/O) programming concepts to create object-oriented applications solving business problems 10%

12. METHODS OF INSTRUCTION (List methods used to present course content.)

1. Discussion
2. Lecture
3. Observation and Demonstration
4. Other (Specify)
5. Projects
6. Critique
7. Multimedia Content
8. Lab

Other Methods:

Use of online libraries from Unity, Microsoft and other vendors

- 13. ASSIGNMENTS:** 6.00 hours/week (List all assignments, including library assignments. Requires two (2) hours of independent work outside of class for each unit/weekly lecture hour. Outside assignments are not required for lab-only courses, although they can be given.)
 Out-of-class Assignments:
 Assigned textbook readings Assigned programming exercises

ASSIGNMENTS ARE: (See definition of college level):
 Primarily College Level

- 14. STUDENT ASSESSMENT:** (Grades are based on):
 ESSAY (Includes "blue book" exams and any written assignment of sufficient length and complexity to require students to select and organize ideas, to explain and support the ideas, and to demonstrate critical thinking skills.)
 COMPUTATION SKILLS
 NON-COMPUTATIONAL PROBLEM SOLVING (Critical thinking should be demonstrated by solving unfamiliar problems via various strategies.)
 SKILL DEMONSTRATION
 MULTIPLE CHOICE

15. TEXTS, READINGS, AND MATERIALS

A. Textbooks:

Albahari, Joseph & Ben Albahari. 2017. *C# 7.0 in a Nutshell: The Definitive Reference* 1st. O'Reilly
 Joe Hocking. 2015. *Unity in Action: Multiplatform Game Development in C# with Unity 5* 1st. Manning Publications
 Watson, Karli, Christian Nagel, Jacob Hammer Pedersen, Jon D. Reid, Morgan Skinner & Eric White. 2008. *Beginning Microsoft Visual C#* 1st. Wiley Publishing / WROX Programmer to Programmer
 Rationale: most current edition; key book in field

*Date is required: Transfer institutions require current publication date(s) within 5 years of outline addition/update.

B. Additional Resources:

Library/LRC Materials and Services:

The instructor, in consultation with a librarian, has reviewed the materials and services of the College Library/LRC in the subject areas related to the proposed new course

Are print materials adequate? Yes

Are nonprint materials adequate? Yes

Are electronic/online resources available? Yes

Are services adequate? Yes

Specific materials and/or services needed have been identified and discussed. Librarian comments:

C. Readings listed in A and B above are: (See definition of college level):

Primarily college level

- 16. DESIGNATE OCCUPATIONAL CODE:**
 C - Occupational

- 17. LEVEL BELOW TRANSFER:**
 Y = Not Applicable

SUPPLEMENTAL PAGE

Use only if additional space is needed. (Type the item number which is to be continued, followed by "continued."
Show the page number in the blank at the bottom of the page. If the item being continued is on page 2 of the
outline, the first supplemental page will be "2a." If additional supplemental pages are required for page 2, they
are to be numbered as 2b, 2c, etc.)

1a. Prerequisites/Corequisites/Recommended Preparation:

RECOMMENDED PREPARATION:

CIS 006: Introduction to Computer Programming
or
CIS 023 C# Programming

;

STUDENT LEARNING OUTCOMES

1. **Outcome:** Create an object-oriented program using the C# .NET programming language.

This outcome maps to the following Institution Outcomes:

- Critical Thinking and Problem Solving - Students will be able to think critically and solve problems by identifying relevant information, evaluating alternatives, synthesizing findings and implementing effective solutions.

Assessment: Evaluate lab assignments and projects to perform Branches and loops of how programs make decisions and choose different actions.

2. **Outcome:** Write valid decision statements (if/else constructs) in the C# .NET programming language.

This outcome maps to the following Institution Outcomes:

- Critical Thinking and Problem Solving - Students will be able to think critically and solve problems by identifying relevant information, evaluating alternatives, synthesizing findings and implementing effective solutions.

Assessment: Evaluate lab assignments and projects for Interpolated strings on how to create a result string for control of the formatting, column width, and alignment of expressions included in the result string.

3. **Outcome:** Create list collection type that stores sequences of data in the C# .NET programming language.

This outcome maps to the following Institution Outcomes:

- Career Technical Education - Students will demonstrate technical skills in keeping with the demands of their field of study.

Assessment: Evaluate lab assignments and projects for List collection type that stores sequences of data. Perform add and remove items, search for items, sort the list, and present different kinds of lists.

Generated on: 5/11/2018 1:15:30 PM