Course Outline

COURSE: CSIS 56  DIVISION: 50  ALSO LISTED AS:

TERM EFFECTIVE: Fall 2011  Inactive Course

SHORT TITLE: GAME PROGRAMMING

LONG TITLE: Game Programming

<table>
<thead>
<tr>
<th>Units</th>
<th>Number of Weeks</th>
<th>Type</th>
<th>Contact Hours/Week</th>
<th>Total Contact Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>18</td>
<td>Lecture: 3</td>
<td>54</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lab: 3</td>
<td>54</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other: 0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total: 6</td>
<td>108</td>
<td></td>
</tr>
</tbody>
</table>

COURSE DESCRIPTION:

This course is an introduction to game programming using Windows game programming tools, using graphics, animation, sound and input devices. The class is a hands-on class where the student will use the basic tools and techniques to create original games. This course has the option of a letter grade or pass/no pass. May be repeated three times for credit. ADVISORY: CSIS 51 Visual Basic Programming or similar programming experience.

PREREQUISITES:

COREQUISITES:

CREDIT STATUS: D - Credit - Degree Applicable

GRADING MODES

L - Standard Letter Grade  
P - Pass/No Pass

REPEATABILITY: R - Course may be repeated  
Maximum of 3 times

SCHEDULE TYPES:

02 - Lecture and/or discussion  
03 - Lecture/Laboratory  
04 - Laboratory/Studio/Activity

STUDENT LEARNING OUTCOMES:

11/6/2012
1. Select game playing hardware and software.
ILO: 3, 2, 1, 5
Measure: Projects, homework, and quizzes
2. Create game programs with sound, graphics, and action.
ILO: 7, 5, 3, 2, 1
Measure: Projects, homework, and tests
3. Create game program libraries that can be used with other games.
ILO: 3, 7, 2, 1, 5
Measure: Projects, homework, and test

CONTENT, STUDENT PERFORMANCE OBJECTIVES, OUT-OF-CLASS ASSIGNMENTS

Inactive Course: 09/26/2011

Students repeating this class will learn features of new software and hardware under guidance of an instructor.

Week 1-2   6/6 Hours
Lecture:
Game programming basics: hardware and software.
Learning Visual Basic
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Select and evaluate software and hardware for developing computer games.
Select the software version they want to use, and start to learn the language.

Week 3-4   6/6 Hours
Lecture:
Visual Basic objects and code
Using Windows Applications Programming Interface (API)
Using API to create points, lines, rectangles, circles, bitmaps, and other objects.
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Write computer games using VB.
Use Windows Applications Programming Interface (API) to create programs.
Create points, lines, rectangles, circles, bitmaps, and other objects.

Week 5-6   6/6 Hours
Lecture:
Designing for efficiency and optimizing code
Object -Oriented programming
Using DirectX
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Write efficient programs and do time tests on the programs.
Write game programs using Object-Oriented programming.
Use DirectX with VB to write more interesting programs.

Week 7-8  6/6 Hours
Lecture:
Bitmap handling in VB and Windows
Sprite properties and functionality
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Use VB and Windows bitmaps.
Write game programs with sprites.

Week 9-10  6/6 Hours
Lecture:
Using high-speed animation
Using DirectDraw
Mid-term projects and quizzes.
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Write game programs using high animation, double buffering and arcade-style scrolling.
Write game programs using DirectDraw surfaces

Week 11-12  6/6 Hours
Lecture:
Introducing sound and music
Handling user input
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Write game programs using with sound and music
Write game programs using user input from the keyboard, mouse, and joystick.

Week 13-14  6/6 Hours
Lecture:
Creating game libraries
using graphics and sprites in 3-D games.
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Create game libraries that can be used with other programs.
Write 3-D game programs using graphics and sprites.

Week 15-16  6/6 Hours
Lecture:
Using artificial intelligence, computer player control, and smart opponents
Using multiplayer programming, sockets, and socketchat.
Homework/Lab:
Read the chapters covered in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Write game programs using artificial intelligence, computer player control, and smart opponent
Write game programs using multiplayer programming, sockets, and socketchat

Week 17  3/3 Hours
Lecture:
Introduction to DirectPlay
Game design basics, development phases, and post-production
Block attack, shooter, and combat games
Homework/Lab:
Read the chapters cover in the class lectures, and do the homework at the end of the chapters.
Do homework and programs assigned in lecture on game programming.
Performance objectives:
Write plans for development and post production of games
Write block attack, shooter, and combat game programs

Week 18  2 Hours
Final projects and test
ASSIGNMENTS:
See content section of course outline.

METHODS OF INSTRUCTION:
Lecture, game examples, sample programs, quizzes, tests.

METHODS OF EVALUATION:
This is a degree-applicable course, but substantial writing assignments are NOT appropriate, because the course primarily:
Involves skill demonstrations or problem solving
The problem-solving assignments required:
Homework problems
Quizzes
The types of skill demonstrations required:
Class performance
Performance exams
The types of objective examinations used in the course:
Multiple choice
True/false
Matching items
Completion
Other category:
None
The basis for assigning students grades in the course:
Writing assignments: 0% - 0%
Problem-solving demonstrations: 40% - 80%
Skill demonstrations: 30% - 60%
Objective examinations: 10% - 30%
Other methods of evaluation: 0% - 0%

REPRESENTATIVE TEXTBOOKS:
"Microsoft Game Playing with DirectX" by Jonathan Harbour Premier Press, 2002, or other appropriate college level text.
Reading level of text: 12 grade. Verified by: dvt
Other Materials Required to be Purchased by the Student: Storage disk

ARTICULATION and CERTIFICATE INFORMATION
Associate Degree:
CSU GE:
IGETC:
CSU TRANSFER:
Transferable CSU, effective 200570
UC TRANSFER:
Not Transferable

SUPPLEMENTAL DATA:
Basic Skills: N
Classification: I
Noncredit Category: Y
Cooperative Education:
Program Status: 1 Program Applicable
Special Class Status: N
CAN:
CAN Sequence:
CSU Crosswalk Course Department: CSIS
CSU Crosswalk Course Number: 56
Prior to College Level: Y
Non Credit Enhanced Funding: N
Funding Agency Code: Y
In-Service: N
Occupational Course: D
Maximum Hours:
Minimum Hours:
Course Control Number: CCC000356482
Sports/Physical Education Course: N
Taxonomy of Program: 070710