

COURSE DESCRIPTION

Department and Course Number CMPS 327 **Course Coordinator** Jim Etheredge

Course Title	Introduction to Video Game Design & Development	Total Credits	3
URL	http://www.ucs.louisiana.edu/~jne1390/cs327/cs327.html	Semester hours	3

Current Bulletin Description

Design, implementation, and testing of video games. Incremental game engine development, graphics, user input, animation, sound, music and artificial intelligence. Sp. Prereq: CMPS 261 and MATH 270/272 with a grade of C.

Textbooks

Beginning DirectX9. Wendy Jones. Thomson Course Technology. ISBN 1-59200-349-4.

Game Design (2e). Bob Bates. Thomson Course Technology. ISBN 1-59200-493-8.

References

None

Course Goals

To gain an understanding of the fundamental concepts of 2D video game design and development. These concepts include game design (both large and small scale), the development and use of a game engine, the use of video game development software, and programming issues unique to video games. Students will learn the basics of DirectX 9 utilization in a Windows/C++ environment.

Course Outcomes

Students will expand their C++ programming skills to include the DirectX 9 graphics package. They will be able use C++ and DirectX 9 to develop basic object-oriented two-dimensional video game components.

Students will have first hand experience in the design and implementation of video games using a popular game development software package. They will design games both individually and in a group, using a standard game proposal document format. They will create/acquire the necessary art and sound resources and implement non-trivial games for review and evaluation by their peers in the course.

Prerequisites by Topic

Students need a familiarity with advanced C++ programming techniques. These are the skills students should possess after successful completion of CMPS260, which is the prerequisite for this course.

Major Topics Covered in the Course

- Introduction to DirectX 9
- Fundamentals of 3D computer graphics
- Matrices, Transforms, and Rotations
- Vertex colors, texture mapping, and 3D lighting
- Meshes
- Particle systems
- Using DirectInput
- Using DirectSound
- The basics of a 3D game engine

Laboratory projects (specify number of weeks on each)

Each week of the course, students learn one of the topics listed above. The assignment for that week is the modification of existing C++ code to demonstrate an understanding of the implementation issues associated with the topic.

Oral and Written Communications

Every student is required to submit at least 2 written reports (not including exams, tests, quizzes, or commented programs) of typically 4-6 pages and to make 2 oral presentations of typically 10 - 15 minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

Social and Ethical Issues

Social and ethical issues are not covered specifically in this course. They are discussed in general as part of classroom discussions of the responsibilities of system analysts and programmers to produce computer systems that fulfill requirements and adhere to accepted legal, moral and ethical standards. Students are not tested directly on these issues. Students are also made aware of the importance of appropriateness in video game development. Students are held accountable in this respect via a severe penalty for the inclusion of any material of a profane, sexual, or violent nature. All student programs are posted on the web site (listed above).

Theoretical Content

Students are introduced to the theory of both 2D and 3D computer graphics. They are exposed to critical issues in the design and development of video games. They are also required to work within a group to develop games that are too large to be developed by a single person in the time allotted.

Problem Analysis

Students in this course are required to perform the analysis necessary to design, create, debug, and test programs to implement the concepts covered. The design process is an integral part of the class since this skill is necessary for computer scientists. Students learn the analysis process through classroom lectures, laboratory work, and guidance provided by faculty and teaching assistants in the understanding and completion of programming and design assignments.

Solution Design

Students in this course are required to design all programming assignments. Assignments are made in the form of requirements documents and the students are responsible for designing the solution prior to implementation and testing.