

Introduction

The game industry is a multi-billion dollar industry and is still growing. Years ago, the technology forced games to have simple designs. Programs could often be developed by a small team of pure hackers with the major requirement being a good idea. Back then, formal training and education was not considered that important. Now, it is almost essential.

Games are so complex these days that they require months if not years to be developed, usually by a large team of programmers, designers, artists, testers, and producers to organize and develop them. Games are now as complex as, if not more than, the latest blockbuster movie. It is an interactive entertainment, pure and simple.

The tools and training needed for game development are enormous, and the complexities warrant academic education beyond a single course. In fact, an entire undergraduate curriculum could be designed and justified for game development.

This course is still experimental and evolutionary in nature. It was offered and shaped in the past by Dr. Christian Duncan and carried out by Dr. Uttam Sarkar. The present structure is from scratch and new at this University, however, it inherits some good ideas from Christian and Uttam. I am grateful to my colleagues for their contributions.

Course objective

Ideally, it would be worth looking at the entire area of game development including story boarding, 3D modeling, sophisticated sound effects etc. but that would take years! Realistically, we would instead take this class as an opportunity to explore the basics behind game programming and related issues befalling game development and gaming industry. While the course is expected to incorporate interests and game development skills from the students, one must not carry the unwanted hope of mastering all it takes to develop a commercial quality game in a semester.

However, the course will focus on necessary topics for creating 2D games. Instead of dealing with existing game engine API's and unwanted infrastructure issues we will explore the available development tools using Java. With Java, we can make fast, full-screen (or applet), hardware-accelerated games while having the benefits of programming in Java with known and sophisticated IDE's (e.g. Eclipse) and lots of open-source libraries.

Pre-requisites and Preparation

The only pre-requisite this course is CSC120 and CSC220 with a strong preference for CSC322 (C programming). The course would best fit students who love programming and who also like do a lot of study and net-searching of game related resources.

Instructor

[Dr. Ubbo Visser](http://www.cs.miami.edu/~visser). Contact details are on the WWW at <http://www.cs.miami.edu/~visser>. The WWW page gives also digital copies of assignments and slides (if not already on the WWW as HTML version). Particular office hours are not given, students who want to talk to me are encouraged to make an appointment with me in class or via email. Students are encouraged to ask questions by email at all times.

Teaching Assistant

Justin Stoecker
Email: jstoecker911@gmail.com
Office: Ungar 440

Contact Hours

Each week there are two 75 minutes sessions (Tuesday, Thursday, 11.00 - 12.15 pm)
Classroom: UB 426

Recommended Text Book

There are a lot of books available with regard to game programming. However, I haven't found a good textbook yet that covers what I would like to do in the course. Thus, I have decided on a Java game programming book rather than a traditional textbook:

- Developing Games in Java by David Brackeen, 2003, New Riders Publishing. ISBN-10: 1-5927-3005-1

Course Content

Course material contains material from various sources reflecting the contents of the recommended book.

Grading

TBA

Scoring of Homework Assignments

TBA

Class attendance and participation

Class attendance is not mandatory, although my exams will depend heavily of my lectures. Not all of the material will come from the text. Class participation is also important. Active interest in lectures is the easiest way to learn.

Plagiarism

The penalty for copied homework of any kind can be immediate failure in the course. My policy on programs is as follows: There is no reason for two (or more) people handing in identical or nearly identical programs. I will regard such programs as either group-written or simply copied. If I have no hard evidence of copying, such programs will receive NO credit. We will do code-checking with the newest programs available. More serious actions will be taken in cases where there is evidence of cheating. More serious actions will be taken in cases where there is evidence of cheating.

Late programs

Unless otherwise stated, programs will lose 20% of their value for each weekday (Monday through Friday) that they are late, down to a minimum value of 20%. The due date of a program is the latest date on which it can be run to get full credit.

Dropping the course

Unless there are extreme extenuating circumstances, I will not allow anyone to drop a course after the drop date. Poor academic performance will never be an acceptable reason for a late drop. The drop date for this course is April 1st.

Incompletes

Unless there has been a documentable illness that caused you to miss substantial amounts of class and computer time, I will not give an incomplete grade in this course. Therefore, please do NOT waste my time asking about an incomplete grade unless you have a remarkably good reason.

Make-up exams

I do not give make-up exams. You simply must show up and take them at the specified times.