

## COURSE DESCRIPTION

DEPARTMENT AND COURSE NUMBER      CMPS 415      **Course Coordinator**      Jim Etheredge

<b>Course Title</b>	Computer Graphics	<b>Total Credits</b>	3
<b>URL</b>		<b>Semester hours</b>	3

**Current Bulletin Description:** Algorithms, analysis, and software architecture for graphical information systems, visualization, realistic rendering, and interactive user interfaces. Project on extensive image representation, transformation, and rendering. Fa. Prereq: CMPS 341, 351 both with a grade of C; or CMPS 405, 406 both with a grade of B.

### Textbooks

Foley, van Dam, Feiner, Hughes. *Computer Graphics: Principles and Practice, Second Edition in C*. Addison-Wesley, 1996.

*Note:* Starting Fall 2007, textbook will be: Hearn, D. and Baker, M., *Computer Graphics with OpenGL (3<sup>rd</sup> Edition)*, Prentice Hall, 2003.

### References

1. Tutorials provided using Moodle or URLs given to students.
2. Introduction to OpenGL by the instructor.
3. Optional textbook: “any good book on OpenGL” such as: Shreiner, Woo, Neider, Davis. *Open GL Programming Guide, Version 2, Fifth Edition*. Addison-Wesley, 2006. (an earlier edition is freely available in electronic form).

### Course Goals

- Understanding of overall architecture of 3D polygon rendering pipeline
- Deeper understanding of each major component of such systems: model representation, scan conversion, illumination, visible surface determination, modeling/viewing/projection transforms
- At least a high-level understanding of other standard graphics techniques: global illumination (ray tracing, radiosity), curves and surfaces, and selected topics such as object dynamics
- Experience programming with a major industry standard graphics library

### Course outcomes

- Students have an understanding of components and their interconnection in a rendering pipeline
- Students are able to write software using an industry standard graphics library such as OpenGL to create scenes with 3D-shaded objects and nontrivial 3D motion
- Students demonstrate an understanding of tradeoffs between different graphics techniques
- Students demonstrate an understanding of the classic graphics algorithms

### Prerequisites by Topic

- Basic knowledge of vector and matrix operations (especially matrix multiplication, dot product, cross product, and vector normalization.)
- C/C++ programming skills and experience from multiple previous CMPS courses
- Working knowledge of development on Unix and/or Windows
- Stated prerequisites: CMPS 341, 351 both with a grade of C; or CMPS 405, 406 both with a grade of B
  - CMPS 341: Foundations of Computer Science
  - CMPS 351: Computer Organization and Assembly Language Programming
  - CMPS 405: Fundamental Principles of Computer Programming

- CMPS 406: Fundamentals of Computing Theory

### Major Topics Covered in the Course

- High-level view of graphics pipeline 1 classes
- Scan conversion 2-3 classes
- Math review for remainder of course 1 class
- Modeling Transforms and pose representations 3 classes
- Model representation and hierarchical organization 1.5 classes
- Viewing and projection 1.5 classes
- Visible Surface Determination 1 class
- Basic illumination (color, lighting, shading, texturing) 2 classes
- Global Illumination 2 classes
- Selected special topic: (recently, an intro to particle dynamics) 2 classes
- Curves and Surface 2 classes
- OpenGL Programming 1 class + discussions throughout course
- History/timeline of computer graphics (using ACM videos) 2 classes
- Exams 2 classes

### Programming projects (specify number of weeks on each)

- Scan convert a 2D primitive (2 weeks)
- Create unlit 3D model with pose controls (translation/rotation) (2 weeks)
- Add moving subparts to 3D model and add lighting/texturing (2 weeks)
- Rudimentary (nonrecursive) ray tracer (2 weeks)
- Draw piecewise cubic curve (or use to control motion) (2 weeks)
- Selected special topic (e.g., bouncing ball simulation) (2 weeks)

### Oral and Written Communications

None (however, when class size is small, each student is required to give one 10-minute presentation)

### Social and Ethical Issues

This is not currently a major topic in the course. However, the impact of computer graphics on our lives is discussed, and academic integrity is repeatedly stressed in the class.

### Theoretical Content

Most lecture content, as listed under “Major Topics” above, is theoretical in the sense that it discusses classic computer graphics methods and algorithms in a conceptual and mathematical manner (estimate 80%).

### Problem Analysis

Understanding the main course content requires problem analysis, guided by lecture (concepts, mathematical models, algorithms, comparison of approaches).

For programming assignments, students must study lecture concepts and apply them based on assignment specifications. The students must analyze the problem, select appropriate approaches, and implement and/or use mathematical components.

## **Solution Design**

Students solve programming assignments based on assignment specifications and concepts discussed in lecture. Students must use efficient approaches, use suitable data structures or object representations, and must implement and/or use mathematical components.