CS 423: Computer Graphics

1. Course number and name: CS 423: Computer Graphics

2. Credits and contact hours: 3.000 Credit Hours, 3.000 Lecture hours

3. Instructor or course coordinator: Bradley Kjell

4. Text book, title, author, and year


   a. other supplemental materials
      - Supplementary material is available on line:
        http://chortle.ccsu.edu/VectorLessons/vectorIndex.html
      - The course uses Blackboard/Vista for course notes, programming exercises, quizzes, grade book, and sample programs.

5. Specific course information

   a. brief description of the content of the course (Catalog Description):
      Wire frame and solid graphics in two and three dimensions, data structures for computer graphics, geometrical transformations in computer graphics, raster, and vector display technologies.

   b. prerequisites or co-requisites: CS 253 or (for graduates) CS 501.

   c. indicate whether a required, elective, or selected elective course in the program: Elective

6. Specific goals for the course

   a. specific outcomes of instruction: Students who complete this course will be able to:
      - write medium-sized graphics programs using basic features of OpenGL
      - explain the basic ideas of 3D computer graphics and the virtual camera model
      - include simple user interaction in their programs
      - solve simple problems using the math used in 3D computer graphics
      - explain basic graphics hardware
      - understand basic imagery human visual perception and digital imagery

   b. Explicitly indicate which of the student outcomes listed in Criterion 3 or any other outcomes are addressed by the course.

      Outcome a: An ability to apply knowledge of computing and mathematics appropriate to the discipline.
      - The course involves extensive use of linear algebra and coordinate transformations and their implementation in a graphics package.

      Outcome b: An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.
The programming problems that students are given require them to analyze them in terms of the operations that are required and how those operations can be specified with OpenGL functions.

Outcome c: An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.

The final project, an interactive mini-world, requires students to analyze their problem, design a solution, and to implement it. Students must coordinate many code modules to achieve their objective.

Outcome d: An ability to function effectively on teams to accomplish a common goal.

In some semesters, one programming assignment requires each student to design a graphical object (such as a house or a car) which is tested, debugged, and put into a library available for all students.

Outcome (i): An ability to use current techniques, skills, and tools necessary for computing practice.

The course involves programming in ANSI C using a modern program development environment (Dev-C++), and using the latest OpenGL libraries.

7. Brief list of topics covered

- Software and hardware components of a graphics system
- OpenGL API, the graphics pipeline, the virtual camera model
- Mathematical review for 3D computer graphics
- Input and interaction
- Geometric Objects and transformations
- Viewing
- Lighting and shading

8. Expected Performance Criteria

Students write biweekly graphics programs of increasing complexity using OpenGL and ANSI C. The programs apply the topics in graphics and programming discussed in class. The final program will create a mini-world through which the viewer can navigate using mouse or keyboard controls. Mathematical and theoretical topics will be exercises through some written assignments and in-class group work. Midterm and final exams assess the mastery of these topics and include some small programs.