COURSE COORDINATOR  Andrew Duchowski

COURSE DESCRIPTION
This is the second of two courses intended to provide technical foundations for DPA students coming from the art or design fields. For undergraduates, it provides a focused introduction to the technical foundations needed for advanced study in digital production. The course will be challenging, and fast paced, and will involve considerable outside work on the computer. The course is also intended for advanced undergraduate or graduate students at Clemson who wish to develop an understanding of the use and programming of visual computing. This course is not intended for students who are Computer Science or Computer Engineering students.

PREREQUISITES & COREQUISITIES
DPA 4000/6000 or knowledge of C++ (or Python) programming, Unix, 2D OpenGL, and approval of instructor.

REQUIRED MATERIALS
Textbook:
Supplemental: OpenGL Programming Guide

COURSE DELIVERY METHOD
Lecture: 3 hours/week
Lab: 0 hours/week

COURSE OUTCOMES
This course is a second technical course on computing for graphics, with the first course being DPA 400. This course builds on this earlier foundation, and completes a comprehensive coverage of the technical and computing background needed for the core courses in the DPA Program. In this course, students will learn advanced topics in C++ (or Python) programming including concepts of object-oriented programming. Projects will all make use of either 2D or 3D OpenGL for the production of interactive graphics. In addition, students will learn the mathematical foundations for computer graphics. This course includes programming projects designed to promote familiarity with 2D and 3D graphics concepts.

4000-LEVEL LEARNING OUTCOMES
(i) Use OpenGL to display a boundary-representation 3D object (e.g., cube).
(j) Use OpenGL to light and shade a 3D object.
(k) Demonstrate use of affine transformations for rotations, translation, scaling.
(l) Use affine transformations to affect camera placement and orientation.
(m) Explain projection transformations.
6000-LEVEL LEARNING OUTCOMES

(n) Use OpenGL to display a boundary-representation 3D object (e.g., cube).
(o) Use OpenGL to light and shade a 3D object.
(p) Demonstrate use of affine transformations for rotations, translation, scaling.
(q) Use affine transformations to affect camera placement and orientation.
(r) Explain projection transformations.
(s) Use suitable math library with vectors.
(t) Use suitable math library with matrices.

TOPICS TO BE COVERED DURING COURSE

The mathematical and algorithmic foundations of computer graphics. Object oriented programming in C++ (or Python), basic mathematics for graphics, 3D graphics API.