

**William Paterson University of New Jersey**  
**Department of Computer Science**  
**College of Science and Health**  
**Course Outline**

1. TITLE OF COURSE AND COURSE NUMBER: Computer Graphics, CS 461,  
Credits: 3, (Major Elective Course)
  
2. DESCRIPTION OF THE COURSE:  
A rigorous study of concepts, theory, and algorithmic aspects of two and three dimensional computer graphics. Topics include: graphics hardware and programming, file-formats, algorithms for curves and image generation, splines, transformations, perspectives, illumination, shading, depth, solid-modelling, ray-tracing, animation, fractalization, texturing, and alternate rendering models. Programming in both general-purpose and ray-tracing languages.
  
3. COURSE PREREQUISITES:  
Prerequisite: CS 342 with a grade of C- or better
  
4. COURSE OBJECTIVES:
  - Teach fundamental concepts, theory, and practical methodologies of computer imaging.
  - Develop student proficiency in deriving mathematical formulas for curves and surfaces.
  - Familiarize students with the hardware aspects of graphics subsystems, internal graphics file-formats and structures, and other pragmatic issues.
  - Build graphics programming experience.
  
5. STUDENT LEARNING OUTCOMES:  
Upon completion of this course, students will be able to:
  - a) Program graphical applications in C++ using OpenGL, ray-tracing, and other languages and API's. This includes rendering images, dynamic generation of images, and synthesis of animation (from program, not real-world sources).
  - b) Understand and program the generation of graphical primitive functions using several algorithmic approaches in the literature. Students will develop the ability to take source code and programming concepts from Internet research sites and several textbooks, monographs, and journals in the library, incorporating that code into their programming repertoire.
  - c) Fully understand and conduct the formulation of mathematical functions, equations, and methodologies needed in computer graphics with a focus on

- algebraic curves first, then parametrics, transformations, graphically-oriented interpolation and spline methods, fractals, space-filling curves, and other areas as they arise in the progression from two to three dimensional modeling. This involves extensive mathematical problem solving with the ultimate production of algorithms and programs.
- d) Identify, understand, and utilize the components of graphics hardware subsystems in computers. They will understand the concept of color in several models and the physics underlying the discipline.
  - e) Visually formulate and design scenes to produce programming language code to generate scenes, including anti-aliasing, 3-D effects (such as shadows, reflection, ...), random-number generators, program-generated texturing, ... In the generation of these programs and their images, students will develop an abstract sense of aesthetics and decision-making relative to it.
  - f) Applying data-structures in the design of efficient programs in computer graphics. Assessment measurement instruments include: exams, surveys, assignments, and projects.

In addition, the course will also reinforce the following students learning outcomes of the university:

- a) Effectively express themselves in written and oral form.  
Measure: exams and assignments/projects.
- b) Demonstrate ability to think critically.  
Measure: exams and assignments/projects.
- c) Locate and use information. Measure: assignments/projects.
- d) Demonstrate ability to integrate knowledge and ideas in a coherent and meaningful manner. Measure: exams and assignments/projects.
- e) Work effectively with others. Measure: group assignments/projects.

## 6. TOPICAL OUTLINE OF THE COURSE CONTENT:

- 1) Fundamentals of Computer Graphics
  - a. Review mathematical structures like matrix algebra, raster coordinates vs world coordinates, parametrics
  - b. Concepts of color, resolution, video modes, and graphics cards.
  - c. Graphics data structures and file formats
  - d. OpenGL in C++
- 2) Fast algorithms for generating geometrical figures such as straight lines, polygons, circles, arcs, and ellipses in OpenGL in C++.
- 3) Two dimensional world
  - a. Windowing
  - b. Clipping algorithms
  - c. Bitmap operations vs vector ones
  - d. Transformations
  - e. 2-Dimensional data-structures and representations
  - f. 2-D Operations in OpenGL in C++
- 4) Three Dimensional Operations
  - a. 3-Dimensional Transformations

- b. Projections and perspective
  - c. Clipping
  - d. 3-Dimensional data-structures and representations
  - e. 3-D Operations in OpenGL in C++
- 5) Parametric Curves & Splines
    - a. Hermite's Method
    - b. B-Splines and other splines
    - c. Bezier Curve Methods
    - d. Nurbs
    - e. Generating Surfaces
  - 6) Light, Color Theory, & Optics
  - 7) L-Systems, Randomization, & Fractalization
  - 8) Hidden Line and Surface Removal
  - 9) Shading Models and Methods
  - 10) Ray Casting
  - 11) Ray Tracing
  - 12) Texturing
  - 13) Animation

7. GUIDELINES/SUGGESTIONS FOR TEACHING METHODS AND STUDENT LEARNING ACTIVITIES:

- a) Classroom lectures, discussions, and problem solving sessions
- b) Homework reviews and exercises
- c) Computer work: programming, experimenting

8. GUIDELINES/SUGGESTIONS FOR METHODS OF STUDENT ASSESSMENT (OUTCOMES):

- a) Periodic examinations and final
- b) Programming projects (10-20 programming projects)
- c) Term paper

9. SUGGESTED READINGS, TEXTS, OBJECTS OF STUDY:

"Computer Graphics using OpenGL", (2<sup>nd</sup> edition), Francis Hill, Prentice Hall, 2001.  
(required textbook)

"Computer Graphics (Schaum's Outline)", (Second Edition, ), Zhigang Xiang and Roy Plastock, McGraw-Hill, 2000. (required textbook #2)

Numerous handouts and on-line tutorials will be distributed on OpenGL, POV-Ray, graphics hardware, and other topics.  
Several websites will be used in each class session.

The following four OpenGL references will be available in the lab for projects & homework reference:

"OpenGL SuperBible" (3<sup>rd</sup> Edition) by Richard S. Wright & Benjamin Lipchak, SAMS, 2004.

"OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 1.4", (4<sup>th</sup> Edition) by OpenGL Architecture Review Board, Addison-Wesley, 2004. Comment: The Redbook – the old, dated edition is on the web.

"OpenGL(R) Reference Manual : The Official Reference Document to OpenGL, Version 1.4", (4<sup>th</sup> Edition), OpenGL Architecture Review Board, Addison-Wesley, 2004.

"OpenGL: A Primer", (2nd Edition), Edward Angel, Addison-Wesley, 2005.

#### 10. BIBLIOGRAPHY OF SUPPORTIVE TEXTS AND OTHER MATERIALS:

"Interactive Computer Graphics: A Top-Down Approach with OpenGL" (4<sup>th</sup> Edition), Edward Angel, Addison-Wesley, 2006.

"Principles of Computer Graphics: Theory and Practice Using OpenGL and Maya", Shalini Govil-Pai, Springer-Verlag, 2005.

"Computer Graphics using OpenGL", (3<sup>rd</sup> Edition), Donald Hearn & Pauline Baker, Prentice Hall, 2004.

"Physically Based Rendering : From Theory to Implementation", Matt Pharr & Greg Humphreys, Morgan Kaufmann, 2004.

"3D Computer Graphics: A Mathematical Introduction with OpenGL", Samuel R. Buss, Cambridge University Press, 2003.

"Real-Time Rendering", (2nd Edition), Tomas Akenine-Moller & Eric Haines, A K Peters Ltd, 2002.

"Fundamentals of Computer Graphics", Peter Shirley, AK Peters Ltd, 2002.

"Level of Detail for 3D Graphics", David Luebke, Martin Reddy, Jonathan Cohen, Amitabh Varshney, B. Watson, & Robert Huebner, Morgan Kaufmann, 2002.

"Texturing & Modeling: A Procedural Approach", (Third Edition), David Ebert, Kenton Musgrave, D. Peachey, K. Perlin & S. Worley, Morgan Kaufmann, 2002.

“Geometric Tools for Computer Graphics”, Philip Schneider & David H. Eberly, Morgan Kaufmann, 2002.

“Practical Algorithms for 3D Computer Graphics”, R. Stuart Ferguson, AK Peters Ltd, 2001.

“3D Computer Graphics”, (3<sup>rd</sup> Edition), Alan Watt, Addison-Wesley, 2000.

"Andrew Glassner's Notebook : Recreational Computer Graphics", Andrew Glassner, Morgan Kaufmann Publishers, 1999.

"Computer Graphics: Principles & Practice" (2nd Edition, C version), James D. Foley, A. van Dam, et al., Addison-Wesley, 1996.

"Interactive Computer Graphics in X", Theo Pavlidis, PWS Publishing, 1996.

"Computer Graphics: An Object Oriented Approach to the Art & Science", Cornel Pokorny, Franklyn Beedle & Associates Inc, 1994.

"Animation: How-to CD", Jeff Bowermaster, Waite Group Press, 1994 (CD).

"Ray Tracing Creations" (Second Edition), Chris Young & Drew Wells, Waite Group Press, 1994 (POV-Ray diskette).

"Power Graphics using Turbo C++", Loren Heiny, Wiley, 1994.

“Advanced Animation and Rendering Techniques”, Alan Watt & M. Watt, Addison-Wesley, 1993.

"Ray Tracing Worlds with POV-Ray", Alexander Enzman et al., Waite Group Press, 1993.

"Morphing on Your PC", David K. Mason, Waite Group Press, 1993.

"3-D Modeling Lab", Philip Shaddock, Waite Group Press, 1993.

Reference set:

“Graphics Gems I”, Andrew S. Glassner, Morgan Kaufmann, 1990.

“Graphics Gems II”, James Arvo, Morgan Kaufmann, 1994.

“Graphics Gems III (IBM Version)”, David Kirk. Morgan Kaufmann, 1994.

“Graphics Gems IV “, Paul Heckbert, Morgan Kaufmann, 1994.

“Graphics Gems V “, Alan W. Paeth. Morgan Kaufmann, 1995.

11. PREPARER'S NAME AND DATE: Dr. John Najarian
12. ORIGINAL DEPARTMENTAL APPROVAL DATE: Spring, 1996
13. REVISERS' NAME AND DATE: Dr. John Najarian, Spring 2005 and previously 4/1/2000
14. DEPARTMENTAL REVISION APPROVAL DATE: Spring 2005