ACADEMIC YEAR 2018-2019
WINTER 2019

Course number	COMP 371	**Course Title**	Computer Graphics
Department	Computer Science and Software Engineering	**Semester**	Winter 2019
Type	Elective	**Credits**	4.0
Level	Undergraduate	**Prerequisites**	COMP 232 or COEN 231; COMP 352 or COEN 352; very good C++ programming knowledge/skills is essential
Schedule	Thu @ 17:45 - 20:15 H 553 SGW	**Lab/Tutorial**	We 4:15PM - 6:05PM, Th 8:30PM - 10:20PM, Th 3:45PM - 5:35PM H831, H845, H843
Instructor	Charalambos Poullis, CSSE	**Office hours**	Wed @ 14:00 - 15:00 and by appointment
Office	EV3.183	**Email**	charalambos@poullis.org
Teaching Assistants	Timothy Forbes Chen [Jocelyn] Qiao	**Email**	comp371.ta@gmail.com

COURSE DESCRIPTION

This course introduces basic techniques and concepts of 3D computer graphics for applications in various sectors which include engineering, visualization, entertainment, gaming, etc. Topics covered include 2D and 3D transformations, modeling and representation, illumination and shading, rendering, texturing, animation, physics-based animation, and the state-of-the art software tools. The student will learn fundamental algorithms and techniques, and gain experience in graphics programming; in particular, how to program in modern OpenGL, a powerful software API used to produce high-quality computer-generated images of 2D and 3D scenes.

COURSE OBJECTIVES

The primary objective of the course is to provide a comprehensive introduction to computer graphics and the related programming principles required when designing and developing computer graphics applications. More specifically the course aims to cover the following:

- Introducing the programming principles and algorithms used in computer graphics
- Designing, developing, testing and debugging of computer graphics applications
- Learning modern GPU programming
- Gaining practical experience with the graphics library, OpenGL (version 3.0 and above).

LEARNING OUTCOMES

By the end of this course, students will be able to:

- identify the core concepts in computer graphics and explain the graphics pipeline i.e. the sequence of steps required to generate a 2D view of a geometrically represented 3D scene (I1.3)
- employ programming principles, data-structures and algorithms of computer graphics for modeling and rendering 3D scenes (I4.3, I4.4, I6.4)
- compare, criticize and assess state-of-the-art techniques in computer graphics (I5.2)
- develop OpenGL software systems for real world applications (I5.1, I6.4)

COURSE CONTENTS

- Introduction to computer graphics and graphics hardware.
- Introduction to graphics API and graphics systems architecture.
- Mathematics of 2D and 3D transformations, and 2D and 3D viewing.
- Color models and basic rendering algorithms.
- Visual realism and visibility.
- Illumination and shading, global illumination techniques, and textures.
- Introduction to curves and surfaces, and 3D object modelling.
- Introduction to computer animation.

TEACHING METHOD

The course comprises of weekly lectures and practical training; both in the form of labs and individual assignments. **It is emphasized that attendance in lectures and labs is mandatory for learning and performing well in this course.**

ASSESSMENT

Assignments (x3)	$5\% + 10\% + 15\%$
Quizzes (x2)	$1 \times 20 + 1 \times 30\%$
Final project (x1)	$1 \times 20\%$

All assignments, quizzes and the final project must be completed in order to pass the class.

Quizzes: Quiz #1 will take place on **February 14th** and, Quiz #2 will take place on **April 4th**.
Assignments/Final project: The goal of the assignments is to gain practical experience in programming 3D computer graphics with OpenGL. There are three programming assignments and one final project. Please refer to the schedule for the assignment descriptions and due dates. All assignments and final project must be completed to pass the course. The assignments may have a small amount of bonus credit. All assignments must be done individually and ran during the lab session for evaluation. Lab instructors will test your knowledge of programming the assignment during this evaluation.
Submission: The assignments should be submitted by 17h00 on the day they are due. Late submissions will be accepted until 3 days following the initial deadline, however, there will be a penalty of 20% from the total assignment grade and [if applicable] no bonus. *The late submission policy does not apply to the project.*

ACADEMIC INTEGRITY POLICY

There is a plethora of online resources for OpenGL and Computer Graphics in general. You are allowed to incorporate code or tips you find on the Web, provided this doesn't make the assignment/project trivial **and** you explicitly acknowledge your sources. You are allowed to discuss assignments with each other, but coding must be done individually.
Please make sure you familiarize yourself with Concordia's Academic Code of Conduct

SUGGESTED REFERENCE TEXTBOOKS

There is no prescribed textbook. Lecture slides for this course will be the primary pointers. There is a vast amount of learning content in the form of notes, programming tutorials, etc. available on the Internet. The following are suggested reference textbooks:

1. OpenGL SuperBible (7th Edition) Comprehensive Tutorial and Reference by G. Sellers, R. Wright, N. Haemel

2. Interactive Computer Graphics (6th Edition) by Edward Angel

COMMUNICATION

This term we will be using Piazza for class discussion. The system is highly catered to getting you help fast and efficiently from classmates, the TA, and myself. Rather than emailing questions to the teaching staff, I encourage you to post your questions on Piazza. If you have any problems or feedback for the developers, email team@piazza.com.
Find our class page here

COURSE SCHEDULE

The list below provides a summary of the material that will be covered during the course as well as a tentative schedule. Labs will support topics covered in the lectures and provide hands-on exercises.

Date	Lecture topic	E.Angel Book chapter(s)	Comments	Tutorial topic
01. Jan 10th	Course Overview Introduction to Computer Graphics	Ch. 1		No tutorials
02. Jan 17th	Input and Interaction Geometric Objects and Transformations	Ch. 3 Ch. 4	Assignment 1 out	Using OpenGL; Hello World example; Triangle example; setting up shaders; model loading (OBJ - vertices only)
03. Jan 24th	Viewing Building Models	Ch. 5		Transformations (rotate, scale, move); coordinate frames; orthographic and perspective camera; model loading (OBJ)
04. Jan 31st	Programmable shaders Lighting and Shading	Ch. 9 Ch. 6	Assignment 1 due Assignment 2 out	Gouraud shading; Phong shading; shaders (interaction)
05. Feb 07th	Shadows with Projections Hierarchical Modeling	Ch. 5.10 Ch. 10		Assignment 1 grading
06. Feb 14th	**QUIZ #1** **Assignment 1 Solution**			Phong lighting model; shadow mapping; shaders
07. Feb 21st	Culling and Clipping Rasterization	Ch. 7.37.7 Ch. 7.8-7.10	Assignment 2 due Assignment 3 out	Cubemaps
08. Mar 07th	Ray Tracing Geometric Queries	Ch. 13.2 Ch. 13.3		Assignment 2 grading
09. Mar 14th	Spatial Data structures Assignment 2 Solution	Ch. 10.5 Ch. 10.10 Ch. 10.12	Assignment 3 due Project is out	Instancing
10. Mar 21st	Texture Mapping Curves and Surfaces	Ch. 8.7-8.13 Ch. 12		Assignment 3 grading
11. Mar 28th	Keyframe animation Assignment 3 solution	Ch. 10.6		Texture mapping
12. Apr 04th	**QUIZ #2** **Project Presentation Rubric**			Splines
13. Apr 11th	Project presentation and evaluation		Project due	No tutorials

GRADUATE ATTRIBUTES

As part of either the Computer Science or Software Engineering program curriculum, the content of this course includes material and exercises related to the teaching and evaluation of graduate attributes. Graduate attributes are skills that have been identified by the Canadian Engineering Accreditation Board (CEAB) and the Canadian Information Processing Society (CIPS) as being central to the formation of Engineers, computer scientists and information technology professionals. As such, the accreditation criteria for the Software Engineering and Computer Science programs dictate that graduate attributes are taught and evaluated as part of the courses. The following is the list of graduate attributes covered in this course, along with a description of how these attributes are incorporated in the course.
Knowledge base: Knowledge of computer graphics and graphics hardware. Introduction to graphics API and graphics systems architecture. Mathematics of 2D and 3D transformations, and 2D and 3D viewing. Color models and basic rendering algorithms. Visual realism and visibility. Illumination and shading, global illumination techniques, and textures. Introduction to curves and surfaces, and 3D object modeling. Introduction to computer animation.
Problem analysis: Use mathematical models as basis for the implementation of problems requiring computer graphics. Analyze the requirements and constraints of the problem in order to determine what design and implementation solutions will be used.

Design: Design and compose computer graphics components involving many aspects such as stated in the course description.
Use of tools: Use specific computer graphics software development APIs to develop elaborated applications, make an educated decision on the tools and APIs to use based on the established requirements, constraints and design.
Individual and team work: Work as a team in the development of an elaborated software development project using computer graphics software and programming tools. Demonstrate and present the project.