ORIGINAL COURSE IMPLEMENTATION DATE: September 2003

REVISED COURSE IMPLEMENTATION DATE: September 2019

COURSE TO BE REVIEWED (six years after UEC approval)**:** February 2018

Course outline form version: 05/18/2018

OFFICIAL UNDERGRADUATE COURSE OUTLINE FORM

Note: The University reserves the right to amend course outlines as needed without notice.

Course Code and Number: COMP 360	**Number of Credits:** 3 Course credit policy (105)

Course Full Title: Computer Graphics

Course Short Title:

(Transcripts only display 30 characters. Departments may recommend a short title if one is needed. If left blank, one will be assigned.)

Faculty: Faculty of Professional Studies	**Department (or program if no department):** Computer Information Systems

Calendar Description:

This course focuses on the development of Computer Graphics technology. Topics include graphics hardware, lighting models, texture models and the geometric representation of shapes and surfaces.

Prerequisites (or NONE):	COMP 251 and admission to the Bachelor of Computer Information Systems degree or the Bachelor of Science with Computing Science major. Note: Students admitted to a CIS or Computing Science minor may register with department permission.
Corequisites (if applicable, or NONE):	None
Pre/corequisites (if applicable, or NONE):	None

Antirequisite Courses *(Cannot be taken for additional credit.)*

Former course code/number:

Cross-listed with:

Dual-listed with:

Equivalent course(s):

(If offered in the previous five years, antirequisite course(s) will be included in the calendar description as a note that students with credit for the antirequisite course(s) cannot take this course for further credit.)

Typical Structure of Instructional Hours

Lecture/seminar hours	45
Tutorials/workshops	
Supervised laboratory hours	
Experiential (field experience, practicum, internship, etc.)	
Supervised online activities	
Other contact hours:	
Total hours	**45**

Labs to be scheduled independent of lecture hours: ☒ No ☐ Yes

Special Topics *(Double-click on boxes to select.)*

This course is offered with different topics:

☒ No ☐ Yes *(If yes, topic will be recorded when offered.)*

Independent Study

If offered as an Independent Study course, this course may be repeated for further credit: *(If yes, topic will be recorded.)*

☒ No ☐ Yes, repeat(s) ☐ Yes, no limit

Transfer Credit

Transfer credit already exists: *(See **bctransferguide.ca**.)*

☒ No ☐ Yes

Submit outline for (re)articulation:

☒ No ☐ Yes *(If yes, fill in transfer credit form.)*

Grading System

☒ Letter Grades ☐ Credit/No Credit

Maximum enrolment (for information only): 35

Expected Frequency of Course Offerings:

Once per 12-18 months *(Every semester, Fall only, annually, etc.)*

Department / Program Head or Director: Talia Q	**Date approved:**	December 2028
Faculty Council approval	**Date approved:**	December 7, 2018
Dean/Associate VP: Tracy Ryder Glass	**Date approved:**	December 7, 2018
Campus-Wide Consultation (CWC)	**Date of posting:**	February 22, 2019
Undergraduate Education Committee (UEC) approval	**Date of meeting:**	March 1, 2019

Learning Outcomes:

Upon successful completion of this course, students will be able to:

- Assess current hardware and software technologies for 3D computer graphics.
- Explain the 3D rendering process.
- Describe geometric models and their corresponding data structures.
- Demonstrate 3D transformations and composite transformations.
- Explain projections and the specification of a view.
- Demonstrate illumination models and texture mapping.
- Implement the behavior and interaction between objects.
- Give examples of advanced animation techniques.
- Illustrate advanced rendering techniques.
- Design and build 3D graphic applications: CAD systems, flight simulators, and interactive games.

Prior Learning Assessment and Recognition (PLAR)

☒ Yes ☐ No, PLAR cannot be awarded for this course because

Typical Instructional Methods *(Guest lecturers, presentations, online instruction, field trips, etc.; may vary at department's discretion.)*

This is a lab-based course based on case studies and extensive development practice. The course will be delivered in lecture-lab format with numerous demonstrations. The lab portion will give students and the instructor the ability to view and interact with current projects.

NOTE: The following sections may vary by instructor. Please see course syllabus available from the instructor.

Typical Text(s) and Resource Materials *(If more space is required, download Supplemental Texts and Resource Materials form.)*

	Author (surname, initials)	Title (article, book, journal, etc.)	Current ed.	Publisher	Year
1.	Gordon, V.S.; Clevenger, J.L.	Computer Graphics Programming in OpenGL with Java	☒	Mercury Learning & Information	2018
2.			☐		
3.			☐		

Required Additional Supplies and Materials *(Software, hardware, tools, specialized clothing, etc.)*

Typical Evaluation Methods and Weighting

Final exam:	%	Assignments:	100%	Field experience:	%	Portfolio:	%
Midterm exam:	%	Project:	%	Practicum:	%	Other:	%
Quizzes/tests:	%	Lab work:	%	Shop work:	%	Total:	100%

Details (if necessary): Grading will be based on the students successfully completing a number of short assignments and a large programming project. Peer evaluation will be employed in grading the group-based term project.

Typical Course Content and Topics

- **Overview of Computer Graphics:** Introduction, Computer Graphics Systems and Related Fields, Java 2D and Java 3D
- **2D Graphics: Basics:** Introduction, 2D Rendering Process, 2D Geometry and Coordinate Systems, The Graphics 2D Class, Graphing Equations, Geometric Models, Constructive Area Geometry, General Path
- **2D Graphics: Rendering Details:** Colors and Paints, Strokes, Affine Transformation, Compositions of Transformations, Transparency and Compositing Rules, Clipping, Text and Font
- **2D Graphics: Advanced Topics:** Spline Curves, Custom Primitives, Image Processing, Creating Fractal Images, Animation, Printing
- **Basic 3D Graphics:** Introduction, 3D Rendering Process, Java 3D API Overview, Java 3D Scene Graphs, The Superstructure, The Nodes, The Node Components, The Structure of a Java 3D Program, Backgrounds and Bounds, Compiling Scene Graphs and Capacity Bits
- **Graphics Contents:** Introduction, Points and Vectors, Geometry, Geometry Info, Primitives, Fonts and Texts, Appearance and Attributes
- **Geometric Transformation:** 3D Affine Transformations, Transformations in Scene Graphs, Composite Transforms, Constructing Geometries with Transformations
- **Views:** Introduction Projections, Specification is a View, Java 3D View Model, Picking, Head Tracking
- **Lighting and Texturing:** Introduction, Lights, Illumination Models, Material, Atmospheric Attenuation and Depth Cueing, Texture Mapping, Texture Coordinates Generation
- **Behavior and Interaction:** Introduction, Behavior, Interaction, Behavior and Picking
- **Animation:** Introduction, Alpha Objects, Interpolators, Morphing, LOD, Billboard
- **Additional 3D Topics:** 3D Curves, Surfaces, Sound, Shadows, Geometry Change, Off Screen Rendering, 3D Textures