Miami Dade College

Course Description

DIG1705 | 3D Programming 1 |4.00 credits

This course, provides students with a foundation in 3D programming which will allow them to develop programs using popular graphics libraries such as DirectX, OpenGL, and GLSL. Students will learn basic image processing, geometric transformations, geometric modeling of curves and surfaces, 3D viewing, shaders, and ray tracing. Prerequisites: COP2335, MAC1105

Course Competencies

Competency 1: The student will demonstrate understanding of math for Computer Graphics by:

1. Explaining foundational concepts of Math for Computer Graphics such as: the difference between a point and a vector, homogeneous coordinates, right hand rule, orthonormal basis, implicit and parametric expressions, and barycentric coordinates
2. Discussing quaternion rotations
3. Modifying and writing programming that uses matrices multiplication for moving, rotating, and scaling objects in 3D

Competency 2: The student will demonstrate an application of 3D graphics programming libraries by:

1. Using industry standard 3D libraries to draw, move, rotate, scale and render 2D and 3D graphics
2. Creating 3D data files for storing simple 3D objects

Competency 3: The student will analyze different 3D coordinate systems for game development by:

1. Creating programs that will read in 3D items from data files and display the items in 3D on the screen
2. Modeling their own 3D items to display in the program they write using an industry standard package (like Maya or 3DS Max)
3. Converting real coordinates to virtual and screen coordinates
4. Discussing different notation for 3D coordinates systems in games

Competency 4: The student will demonstrate an understanding of graphics programming for standard development engines by:

1. Creating a project in Unity (or other industry standard engine) and using the raw interface to the engine's drawing functions to create content through scripts alone
2. Writing a shader for Unity (or other industry standard engine)
3. Creating a project in Unity (or other industry standard engine) where a raycast is used to detect line of sight and make angular calculations

Competency 5: The student will demonstrate understanding and application of rendering methods by:

1. Summarizing articles about new and current 3D rendering hardware and software on the market
2. Planning a 3D rendering pipeline
3. Summarizing the differences between graphics libraries and the rendering capabilities of an industry standard engine (like Unity or Unreal)
4. Using a wrapper to extend the functionality of an industry standard engine (like Unity or Unreal)

Competency 6: The student will demonstrate an application of different drawing and programming methods for curved surfaces by:

1. Creating 3D objects that have curved surfaces and adding them to their 3D worlds of objects
2. Discussing papers on different ways to program curved surfaces
3. Using curved surfaces in a final project

Competency 7: The student will demonstrate an application of 3D graphics programming development:

1. Working in a group on a final project in Unity (or another industry standard engine) which simulates a virtual world and employs the following elements in the simulation: procedurally drawn geometry from code (including at least one curved surface), a custom shader, ray tracing, raycasting, and quaternion rotations
2. Creating a final project that can take the form of a game or a virtual world

Learning Outcomes:

- Solve problems using critical and creative thinking and scientific reasoning
- Formulate strategies to locate, evaluate, and apply information
- Use computer and emerging technologies effectively
- Demonstrate an appreciation for aesthetics and creative activities