**CMP3103 Computer Games Development**

| Period per Week | | | Contact Hour per Semester | Weighted Total Mark | Weighted Exam Mark | Weighted Continuous Assessment Mark | Credit Units |
|---|---|---|---|---|---|---|---|
| LH | PH | TH | CH | WTM | WEM | WCM | CU |
| 45 | 30 | 00 | 60 | 100 | 60 | 40 | 4 |

**Rationale**

Computer games provide a contemporary gaming and entertainment platform as well as stimulating the mind of the developer toward high level of creativity. It is therefore imperative for the computer engineering student to be abreast with the fundamentals of computer games development. Using a comprehensive and analytical approach to game engine architectures, this is course designed to offer students the opportunity to learn the effective implementation game ideas, algorithms and graphics.

**Objectives**

The objectives of the Computer Games Development course are to:

- Provide students with the core programming language skills necessary for game development.
- Provide students with a thorough understanding of the main concepts involved in real-time 3D graphics programming.
- Provide students with a fuller understanding of current and emerging digital entertainment systems technologies
- Facilitate students in the development of expertise and interest in topic areas of digital entertainment systems
- Provide students with an understanding of current digital entertainment systems research issues
- Further develop students' analytical, creative, problem- solving and research skills

**Course Content**

1. ***C++ Programming for Game Developers***
   - Template Classes and Template Functions
   - Error Handling
   - Number Systems, Data Representations, and Bit Operations

- The Standard Template Library
- Introduction to Windows Programming
- Menus and Drawing with GDI
- Dialog Boxes
- Timing, Animation, and Sprites
- Designing and Implementing a 2D Game

2. ***Graphics Programming with DirectX***
   - 3D Mathematics
   - The Transformation and Lighting Pipeline
   - Initializing Direct3D
   - Vertex and Index Buffers
   - Single and MultiTexture Effects
   - Camera Management Systems
   - Compressed Textures
   - Texturing & the Texture Blending Cascade
   - Alpha Blending
   - Loading GILES™ Scenes
   - The D3DX Mesh Containers
   - Progressive Meshes
   - Scene Hierarchies
   - Keyframe Animation
   - Indexed / Non-Indexed Vertex Blending
   - Software and Hardware Skinning Techniques
   - Skeletal Animation
   - Tree Animation and Rendering
   - X Files
   - Motion Blending
   - Collision Detection and Response
   - Quadtrees, Octrees, and kD-Trees
   - Binary Space Partitioning (BSP) Trees and Potential Visibility Sets (PVS)

**Learning Outcomes**

On completion of this course the student should be able to:

- Implement a game algorithm using C++.
- Write a game script
- implement game graphics, animations and voicing using state-of-the-art tools and technologies
- demonstrate a firm understanding of current digital entertainment systems research issues

**Recommended and Reference Book**

*[1]* Michael Dawson, 2004. *Beginning C++ Game Programming (Game Development Series.* Course Technology PTR; 1 edition. ISBN-10: 1592002056,ISBN-13: 978-1592002054

*[2]* Frank D. Luna, 2003. *Introduction to 3D Game Programming with DirectX 9 (Wordware Game and Graphics Library.* Wordware Publishing, Inc.; 400 edition. ISBN-10: 155622913, ISBN-13: 978-1556229138