Due Dec. 14, 2006

In the term project, you will build a game engine and demonstrate its capabilities in a game of your choosing. Following are some guidelines you must follow to accomplish this:

**Project Goals**

- Your focus on this project is on building a game engine, not the game itself. This means
  - Evaluation of the project is on demonstration of the capabilities of your engine, not the game content itself.
  - Hence, your content can come from external sources, can be generated procedurally and stored, or through interactive tools, etc.
  - The content should be chosen to illustrate the capabilities and limitations of your game engine.
  - Exploit the capabilities of OpenGL and the graphics card to maximize the performance of the engine.

- Use the knowledge and experience you have gained in previous course projects and bring all of this into the term project.

- You have limited time on this project (about 4-5 weeks); make the needed compromises, but justify these in detail on your demo and report.

**Game Engine Components**

Your game engine should, in general, contain the following components (not an exhaustive list):

- **Model Representation**: How you store, access your game content (polygonal geometry, textures, etc) can influence performance; typically use of some spatial data structure, bounding volume structures or scene graph is necessary; complex tools such as scene graphs, BSP trees might require use of existing libraries.

- **Collision Detection**: Will need to have some means to detect collisions between game characters, autonomously moving objects and the environment.

- **Richness in Content**: Use of textures, sprites (in complex outdoor environments, for instance) can also place special stress on game engines.

- **Object Culling**: Culling algorithms can significantly improve performance, especially with large amounts of geometry.

- **Hardware/Software Choices**: Current graphics cards permit various traditional CPU operations to be done in hardware. Game engines can take advantage of these.
♦ Maintaining Frame Rate: Choices/compromises made to maintain frame rate, including Level of Detail, rendering compromises, etc. Use of profiling tools to demonstrate performance.

Software Design

- This is a team project. Need to organize the different components of the projects, software style, class hierarchies, etc.
- Decide on each person’s role/responsibility, have self appointed deadlines.
- Consider using CVS, make frequent backups. Loss of software will affect entire team’s effort.

Evaluation/Requirements

- Project will be performed in teams of 3 students each.
- Project will be demonstrated by each team during a part of the final exam (May 14): 30 minute presentation/demo for each project.
- Peer evaluation will be used as part of the overall evaluation.
- Each team will turn in a project report, that will in detail describe the overall design and implementation, goals of the engine, and how well the goals were met.