Modern Computer Games
COMP 521, Winter 2015

Project Description

Description

The project is intended to allow you to explore topics related to the course content that were not well covered by the assignments, or perhaps even the course content. This is expected to be a system-oriented project, and your project should involve significant implementation and experimentation. Note that your project does not need to result in a complete game, nor does it have to include significant effort on visual aesthetics such as textures, models, effects (unless they are part of the goal of your project).

Consider the topics discussed in the course, listed in the syllabus, or which you feel are interesting in a game context. Choose **two** topics and design an **experimental investigation** that demonstrates and/or evaluates the ideas discussed. Your project should include non-trivial complexity in terms of overall scope/design, a significant implementation component, and an appropriate amount of experimental evaluation and analysis.

A non-exhaustive list of suitable topics includes:

- Narrative analysis
- Map/level generation (other than purely as mazes); e.g., city/town structure generation, interiors of buildings, level design
- Alternative or advanced game physics (other than projectiles)
- Detailed/extended use of Verlets
- Collision handling: (hierarchical) bounding box approaches, approximate detection, resolution
- Terrain analysis: dominance, choke-points, reachability measures, cover/strategic positions
- Path-finding: hierarchical approaches, generating waypoint maps and/or navigation meshes
- Visibility: weak/strong forms, 3D forms
- Game AI: goal-oriented designs (GOAP, HTN)
- Groups: advanced steering behaviours
- Procedural content generation
- Game balance/pacing
- Networking: client/server, P2P, consistency, dead-reckoning, scalability issues
- Mobile gaming development and concerns (not covered in the course)
- Sound (not covered in the course)

Your project can be done **individually** or in **groups of 2.** Groups of 3 or more are not allowed. When working in pairs, the efforts of each person should be clearly distinguished, and your overall design should attempt to minimize dependencies. I strongly suggest you also structure your project in terms of layers of increasing complexity—make sure there is a basic core you can certainly accomplish!

You may use any implementation context, but must only use freely available tools/resources. Your code should run on Unix, Windows 7/8, or Android (mobile). (Macs are discouraged, as the instructor does not have access to one.)

Specific Requirements and Deadlines

1. **Tuesday, March 24, 2015, 6pm**: Submit a brief proposal, no more than 1-page, with a high-level description of what you will do. This should clearly describe the two major topics on which you will be basing your project, the general goal/idea, and for groups, what each person will be doing and how you will minimize dependencies between people. A group only needs to submit one proposal for the group.

   Note that the proposal is used to verify the scale/appropriateness of your project, giving you feedback on such if necessary, but will not be graded.

2. **April 7–14, 2015**: In lieu of the last week of classes, brief demonstrations of projects will be held. The precise dates, times and durations are TBA, but plan on having the project sufficiently complete to present by Thursday April 9.

3. **Tuesday, April 14, 2015, 6pm**: Submit a a final report describing your project in detail. The report should be 5–12 pages not including references/title-page, and include the following:

   (a) Cover page: name(s) and project titles.

   (b) Introduction: state the main objectives of the project, briefly describe the methodology, and summarize the results. For those working in groups, separately and clearly identify what was done by each person in terms of the project content, and also the report itself.

   (c) Background: give background on the problems you are addressing. This should include at least 2 technical references (by which I mean academic books/papers or detailed industry explications, and not just websites/blogposts).

   (d) Methodology: give details on the project implementation and design. This should state with an overview, and then use subsections to expand on details.

   (e) Results: describe the experimental context/strategy, and give numerical (tables and/or graphs) data summarizing your results. Include discussion of anomalies, trends, and any other interesting observations you can make on the data.

   (f) Conclusions/Future work: briefly summarize the project, and describe interesting/useful directions in which your work could/should be extended.

   (g) References: formal citations for any referenced works.

   Figures/screenshots in the report can be helpful ways of explaining/describing ideas and as such are encouraged, but the bulk of your report should be a written explanation. A group only needs to submit one report for the group.

   Source code/builds should be separately provided.

Evaluation

The project is worth a total of 30% of your final grade (25% for the project itself, and 5% for the presentation/demo). The project evaluation will be based on the following considerations:

- Choice of specific problems: complexity, and difficulty, relevance to games.
- Approach: appropriateness and complexity of design, consideration and investigation of suitable algorithms/techniques.
- Implementation: difficulty, skill demonstrated,
- Experimentation: appropriateness for the topic/objective, awareness of confounding factors, possible parametrizations, variance, presentation.
- Overall: level of completion, scope, novelty, demonstrated understanding.
- Code: must be working, bug-free, reasonably efficient, and have a professional style.
- Demo: working, coherent and organized.

Expectations for groups are greater than for individuals. Both members of a group must participate in the demo.

Example Topics

The following are a few example ideas for projects to use as starting points and give you an idea of scope. You may use these if you wish, although these are very high-level/abstract and you will still need to describe the goal in more detail and define specifics of your intended approach and evaluation to form a project proposal.

1. Steering behaviours and AI: Based on an adaptation of an existing or fabricated game context, explore the use of different composite steering behaviours to enact goal-oriented group (or individual) competitive strategy.

2. Strategic waypoints and visibility: Using game levels/terrains reproduced based on one or more existing games, geometrically analyze visibility (or movement or other) properties to automatically place cover or other kinds of waypoints under variable player/enemy placement scenarios.

3. PCG and terrain generation: Define and implement a process for generating non-trivial building interiors with some measures/guarantees of quality and/or appropriate structure.

4. Networking and path-finding: Use different path-finding approaches to predict movement and thus improve the accuracy of dead-reckoning.