CS 251: Box2D Project Details

- Handed out: Early in the semester. Due: Oct 20 11pm (don't forget latedays)
- Please write (only if true) the honor code. If you used any source (person or thing) explicitly state it. You can find the honor code on the web page.

Overview

In this project, we aim to build, in Box2D a simulation of just about anything that Box2D allows and is sufficiently complicated as to deserve merit. Use the Box2D version given to you in class so that there are no compilation mismatch.

The Tasks

Out of the tasks mentioned below, the report and the actual implementation carry the most weightage towards the marks for this assignment. Recall that this assignment is worth 15%

1. `[Version Control]` You must have been maintaining a git repository of your project in an area owned by your group. You have to submit this repository finally for us to be able to view your progress in the project.

   We will check out (clone) your project from this repo. That will help us generate all the stuff below.

   This is the job of the program manager (CEO in a startup).

2. `[Makefile]` Make sure you have a makefile that has targets to cleanly compile your project taking care of all dependencies, clean up the project folder, compile your report, and generate documentation. Specifically have the following targets in your makefile.

   (a) codeDoc
   (b) report
   (c) clean
   (d) distclean
   (e) profile
   (f) release

   See details below on each of the above parts and more. There will be some overlap between parts, but try to minimize.

   This is the job of the software guru (CTO in a startup).

3. `[Project Webpage]` Make a webpage for the final project. Have a brief description of what your project is all about on the webpage. Feel free to add screenshots to the page. Similar to what you have been doing for your labs, link this webpage to your `public_html` pages.

   This is the job of the public relations manager (CEO in a startup). Make sure you have a video of your project. This should be no more than 3 minutes. Average 2 min. 640x480 acceptable, HD preferred. I recommend voice. It makes a huge difference. Don't worry about your accent. (However, write down your text before you speak otherwise you will have a tough time in video

production. If you don't do voice, add audio – but be conservative, don't do rock or metal. "Elevator music" is possible. If you can't do audio, do text subtitles.)

Do not link your source code to the Project page.

You can use whatever you want in making beautiful html (i.e. reuse web templates.) However do not use stuff that the authors have forbidden you to use (i.e. use GPL, open source versions). What is not acceptable is to borrow from the Internet and claim it as one's own.

4. `[Code Documentation]` Document all the changes you make to the base code suitably and as thoroughly as you can with Doxygen.

   The goal is to help a programmer (not to help a reviewer of your project). The reviewer of your project will look at this section, but will pay greater emphasis on the main report.

   This is the job of the software guru (CTO in a startup).

5. `[Profiling]` Use `gprof` to profile your code and identify the parts that take significant portions of the time. Try optimizing the code if you can and report that. Also, identify the differences from the profiling data obtained from the original base code. Try explaining these based on the changes you made. You would have to document this in the project report described below.

   This is the job of the software guru (CTO in a startup).

6. `[Report]` Write a report describing your original project idea, what you actually implemented and your reasons for the deviations, if any. Ideally you should include a SRS (See `http://en.wikipedia.org/wiki/Software_requirements_specification` and `http://www.tricity.wsu.edu/~mckinnon/cpts322/cpts322-srs-v1.doc`

   Include your observations regarding the profiling obtained. Make the report interesting by adding project screenshots and plots and graphs wherever possible

   Do add appropriate references to your project using BibTeX.

   Do mention techniques that you have learnt in the lab. However, do not artificially perform something simply because of this instruction – not all the things you learnt in the course may be directly useful in the project.

   Do not forget the honour code.

   Do mention who has done what and how much.

   Also, include in your report a section on what were the difficulties you faced in your project and how you overcame them, maybe by employing the skills learned in the course. Mention any other interesting points about your project too.

   The LaTeX documentation will give the entire Ramayana katha (i.e. the entire story) of the project. In addition, you should have an executive presentation also that summarizes the key points. There will be no more than 5 slides in the presentation and it will play for no more than 3 minutes. The CEO will be using to talk to the "angel" investor.

What to Submit

Your git repo. The readme which includes nothing except number of latedays taken, the honor code, and the distribution of weights for the credits. All other info such as bibliographic citations should appear in the report. The git repo will not contain the project html web page, or the video (unless you can fit it in the 10MB limit of ASC.)

Put everything in a folder. The folder and its compressed version should both be named `lab12_groupXY_final`. Hence, you submit a tar.gz named `lab12_group07_final.tar.gz` if your group number is 7. Submit it under Lab 12 in Moodle.