Final Project

Deadlines for final project

Week I  Nov. 10,11 Literature survey, papers; 2-5 page proposal writeup including conceptual definition of intended example

Week II  Nov. 17,18 Example interface

Week III Nov. 22 Example implementation, presentation outline

Week IV Dec. 1 Presentation paper due

Week V Dec. 3-10 Presentations to the class

Guidelines

- The literature survey should include at least 3 references to books and articles. It will be necessary to present the papers and books (or copies therefrom) that you will be using. At least one of them has to be to a scientific journal, and one should be different than the ones provided by the instructor (if any). You may use the following resources to help you with the bibliographic search:
  
  - ACM digital library: http://www.acm.org/dl You may register and create a non member account. The search engine is very powerful and allows you to create binders on different subjects. You may find the articles by locking them up in the library, and if you do not succeed you may purchase it online. If you consult with the instructor, you may be reimbursed for the article purchase.
  
  - IEEE Computer society online: http://www.computer.org/publications/dlib/index.htm Access here is more restrictive, but you will be able to find titles and abstracts.

- The conceptual definition of example will describe an example that illustrates the general area of research for the final project.
Project topics

The following are possible topics. Topics marked with a star have been or will be discussed in class. The list is by no means exhaustive:

1. **Distributed databases**: implement a linear hashing scheme that simulates a distributed database where nodes can be added and deleted. Linear hashing has the advantage that allows the number of has entries to contract and expand gracefully as items are inserted and deleted. References can be found on Knuth 1998c, pp. 548–549.

2. **Persistent data structures**: these are data structures that represent changing information in such a way that past history can be reconstructed efficiently. Example of applications include revision control systems and undo functions. References can be found on Knuth 1998c, p. 583. You are to implement a persistent binary search tree as an example of such a structure (Cormen and Leiserson 1992, pp. 278–279).


5. *Graph algorithms 2: Applications in biomolecular dynamics.* References: instructor.


13. **Computer graphics**: Illustrates use of inheritance. **3D rendering**, using OpenGL.

14. **Stack applications** Postscript file generator. Examples: **Tree/planar graph drawing program**.


16. **Limits to computation**: **NP-completeness**.