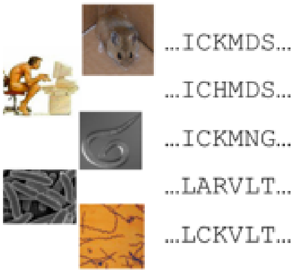
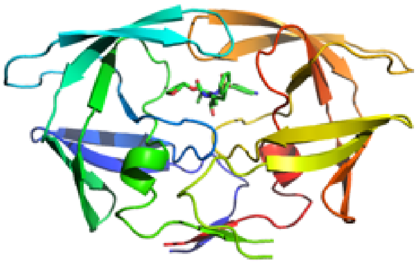
CS 75/175, Winter 2012

Intro to Bioinformatics

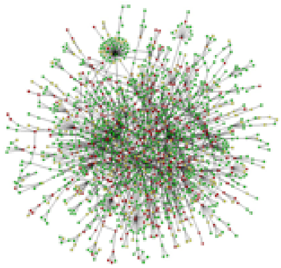
About Schedule



sequence similarity across organisms



structure of HIV-1 protease + drug [1hpv]



possible *functional* interactions [Jeong+]

Course description

Computation is vital for modern molecular biology, helping scientists to model, predict the behaviors of, and control the molecular machinery of the cell. This course will study algorithmic challenges in analyzing biomolecular *sequences* (what genes encode an organism, and how are genes related across organisms?), *structures* (what do the proteins constructed for these genes look like, and what does that tell us about their mechanisms?), and *functions* (what do these things do, and how do they interactw ith each other in doing it?). The course is application-driven, but focused on the underlying algorithms and information processing techniques, employing approaches from search, optimization, pattern recognition, and so forth.

For a beautiful visualization of the underlying molecules, see *The Inner Life of the Cell*, with either musical accompaniment or narration.

To get a feel for the content (although it will be revised), see the previous version.

Administrative info

Instructor
Gevorg Grigoryan | 113 Sudikoff | office hours: ThF 4, or by appointment

Teaching Assistant
Dr. Jian (James) Zhang | 210 Sudikoff | office hours: TuTh 1

Announcements and discussion board
Available on Blackboard

Lectures
2A Period | TuTh 2:00-4:00 | 028 Haldeman Center

Textbook
None required. While there are several great books out there covering some of the material, I haven't found one that comprehensively introduces the topics we'll cover (biomolecular sequence, structure, and function) from a computational perspective. Thus I'll distribute lecture notes and provide references to the literature.

Coursework

The course seeks to provide broad exposure to some important algorithmic challenges and approaches in bioinformatics. The lectures and readings will provide a fairly general (for a ten-week course) survey of the field. As this is an introductory course targeted at undergraduate and graduate students in computer science, a background in biology is not required. However, students should be interested in learning some basic molecular biology and biochemistry, in context.

Students from other related disciplines are encouraged to take the course; it has been approved as an elective for MCB students and to fulfill the applied math requirement for Thayer students. Naturally, a basic understanding of the underlying computational techniques is expected. The homework assignments require implementing bioinformatics algorithms, so programming experience is necessary. Please contact me to discuss your background and interests.

The course is dual-listed as CS 75 for undergraduate students and CS 175 for graduate students. Grades will be handled separately, and each assignment specifies additional requirements for graduate students.

The coursework consists of the following:

Homeworks (70%)
Since one of the best ways to learn something is by doing it, weekly programming labs for roughly the first two-thirds of the term will offer the opportunity to implement and apply some core bioinformatics algorithms. To standardize and enable us to provide some infrastructure, labs will be done in python (version 3.1); students have reported that learning/practicing python is another valuable outcome of the class.
In addition, graduate students will have the chance to practice their skills at critically reading primary literature, with weekly short critiques and discussion of papers related to the topics covered in class.

Final project (25%)
The final project provides the opportunity to more deeply explore a topic of interest, individually or in a group, choosing an appropriate mixture of research, implementation, and application. A project proposal and a project update will ensure that we are on the same page; a project presentation will make for a fun class-wide exchange; a project report will document what was done and what was learned.

Participation (5%)
This is some encouragement to participate in class (asking questions, pointing out potential pitfalls, characterizing strengths, etc.) as well as outside of class (suggesting examples, pointing out significant applications and recent developments, finding related materials, etc.).

Assignments are to be turned in via Blackboard *before* class on the due date. Ample time is provided for each assignment, so in general, late submissions will not be accepted. Under extenuating circumstances, arrangements must be made with the instructor *before* the due date.

Flu policy. In keeping with Dartmouth's flu response effort, *do not come to class* if you have flu-like symptoms. Contact me, and follow the prescribed protocol. You do not need to do any sequence, structure, or function analysis of your flu strain to be excused.

Honor code

Dartmouth's honor code applies to this course, and academic misconduct policies will be strictly enforced. If you have questions, ask!

You may discuss the programming labs with other current CS 75/175 students, but your submission must be entirely your own. As part of a discussion, you may show (in person) another student your work. However, your submission must be created and documented by you alone. You may not copy anything directly from another student's work. For example, copying a portion of someone else's solution onto a piece of paper would violate the honor code, even if you eventually turn in a different answer. Similarly, e-mailing a portion of your code to another student, or posting it on-line for them to see would violate the honor code. Although all students must create and type in their own code, you may help other students debug their programs once you and they have already written your programs. Discussion of the labs is encouraged, subject to these rules; such discussion will be most useful when both students have already made serious attempts to solve the problems on their own.

If you make use of any code that is not your own, you must clearly attribute the source of the code with clear comments in the code that you submit. You do not need to acknowledge discussion with other students on your submitted work. Proper respect for copyright laws as applied to printed materials and software products is subsumed by Dartmouth College's Computing Policies.

Written paper critiques are to be done individually; class time will be reserved for discussion after submission.

The final project may be conducted individually or in a group. Here, collaboration is encouraged. Building on existing code is acceptable, subject to the above comments regarding attribution.

Disabilities

Students with disabilities enrolled in this course and who may need disability-related classroom accommodations are encouraged to make an appointment to see the instructor before the end of the second week of the term. All discussions will remain confidential, although the Student Accessibility Services office may be consulted to discuss appropriate implementation of any accommodation requested.

Last modified: Wed Dec 21 15:39:06 EST 2011