

CSE3800/BME4800/CSE5800: Bioinformatics – Fall 2016

Lecture: TuTh 3:30PM-4:45PM ITE 119



Instructor:

Ion Măndoiu
ion@engr.uconn.edu
Office: ITE 261
Office Hours:
Tu/Th 5pm-6pm
We 12:30pm-2pm

Course Description: This course is an introduction to the fundamental mathematical models and algorithmic techniques used in bioinformatics. Emphasis will be placed on modeling computational problems arising in biology as graph-theoretic, statistical, or mathematical optimization problems, and on designing, analyzing, and implementing efficient combinatorial algorithms for the latter. Covered algorithmic techniques will include exhaustive search, integer programming, greedy algorithms, dynamic programming, divide-and-conquer, graph algorithms, combinatorial pattern matching, clustering, hidden Markov models, and randomized algorithms. Biological applications covered will include restriction mapping, DNA sequencing, motif finding, pairwise and multiple sequence alignment, gene prediction, evolutionary trees, and genome rearrangements.

Prerequisites: BIOL 1107, CSE 1100 or 1010, and either STAT 3025Q or STAT 3345Q

Textbook:

- Neil C. Jones and Pavel A. Pevzner, *An Introduction to Bioinformatics Algorithms*, MIT Press, 2004. Electronic version available (on campus network) at <http://www.netLibrary.com/urlapi.asp?action=summary&v=1&bookid=125977>

Grade breakdown:

In-class quizzes	10%
Theoretical homework assignments	20%
Programming assignments	30%
Final project	40%

Assignment submission: Solutions for theoretical homework assignments must be submitted in electronic format via Moodle. Programming assignments must be submitted electronically via the Rosalind site at rosalind.info. Rosalind is a repository of intellectually stimulating problems of varying difficulty that are extracted from real challenges of molecular biology. Solutions can be prepared using any high-level programming language. You will be asked to process a dataset generated by Rosalind on your own computer and then upload or copy-paste the solution to Rosalind along with your source code. Each submitted solution is automatically checked for correctness, allowing you to fix potential problems before the due date. To access programming assignments you must first enroll for the appropriate Rosalind class using one of the enrollment links below:

- CSE3800: <http://rosalind.info/classes/enroll/332f70c594/>
- BME4800: <http://rosalind.info/classes/enroll/143de85b11/>
- CSE5800: <http://rosalind.info/classes/enroll/9887027280/>

Late policy: All assignments are due at midnight on the specified due date. Late submissions for homework and programming assignments are allowed for up to three days after the original submission deadline, but

10% of the grade will be subtracted for each late day. The lowest homework assignment score and lowest programming assignment score will be dropped from the overall grade calculation.

Final project: The final project, typically done in teams of 2-3 students, will give you the opportunity to study a bioinformatics problem in more depth. Suitable final project topics include surveys of bioinformatics topics not covered in the lectures, design and implementation of novel algorithms, and empirical comparisons of existing methods. A list of potential topics will be provided towards the middle of the semester, although you are strongly encouraged to devise your own. Project requirements will include submitting two intermediate progress reports and a written final report of 15-20 pages. You will also be required to give a short presentation on your project at the end of the semester.

Moodle site: Course materials, assignments, grades, and announcements related to the class will be distributed using Moodle. To get access to the moodle site you must first create a new account at <http://dna.engr.uconn.edu/moodle/> then self-enroll for the bioinformatics class at <http://dna.engr.uconn.edu/moodle/enrol/index.php?id=8> using enrolment key "BIOINFO". You can also use Moodle to ask class-related questions and communicate with your peers and the instructor. Please observe basic etiquette by keeping your postings polite, concise, and on-topic. Before posting do take a look at the postings that are already there—it is possible that your question has already been answered. Appropriate questions are general questions about the material and clarifications on the assignments, for questions that are specific to your own work you should contact the instructor directly.

Academic honesty: You are expected to adhere to the highest standards of academic honesty. All submitted solutions must be your own work. For homework assignments and programming projects you may discuss ideas and concepts with others, but must not share written solutions or code. Use of published materials (including web resources) is allowed, but all sources should be explicitly acknowledged in your solutions. Violations will be reviewed and sanctioned according to the University Policy on Academic Integrity.

Students with disabilities: If you have a documented disability for which you are or may be requesting an accommodation, you are encouraged to contact the instructor and the Center for Students with Disabilities or the University Program for College Students with Learning Disabilities as soon as possible to better ensure that such accommodations are implemented in a timely fashion.

Tentative Schedule:

Week #	Lecture dates	Topics and reading assignments
1	Aug 30 & Sept 1	Intro to molecular biology (Ch. 3)
2	Sept 6 & 8	Exhaustive search (Ch. 4), integer programming (handout)
3	Sept 13 & 15	Greedy algorithms (Ch. 5)
4	Sept 20 & 22	Dynamic programming (Ch. 6)
5	Sept 27 & Sept 29	Dynamic programming (Ch. 6)
6	Oct 4 & 6	Divide and conquer (Ch. 7)
7	Oct 11 & 13	Graph algorithms (Ch. 8)
8	Oct 18 & 20	Graph algorithms (Ch. 8)
9	Oct 25 & 27	Pattern matching (Ch. 9)
10	Nov 1 & 3	Pattern matching (Ch. 9)
11	Nov 8 & 10	Hidden Markov models (Ch. 11)
12	Nov 15 & 17	Hidden Markov models (Ch. 11)
13	Nov 29 & Dec 1	Clustering and phylogenetic trees (Ch. 10)
14	Dec 6 & 8	Randomized algorithms (Ch. 12)