STEM MUSE + SAGES PRESENT

STEM MENTORSHIP PROGRAM!

- Serves women & underrepresented-gendered people in STEM: undergrads, grad students, research staff, postdocs
- One-on-One mentorship: Mentees choose mentors
- Opportunity to connect with STEM companies & mentors
- Networking and professional development opportunities: group mentorship, workshops, socials, and resume/CV-bolstering events
- Flexible scheduling: pairs set schedule at kick-off (1 mtg./mo. for 6 months from February to July)
- Option for in-person or virtual interactions
- Conversation guides facilitate productive conversations

Testimonials: “10/10 would recommend”, “best and most organized”, “helped me figure out where I wanted to go and how to get there”

SCAN THE QR CODE TO SIGN UP AS A MENTEE AND/OR MENTOR

Mentors from more than 60 companies participated in our past programs (2020-2022), including:

BCH394P/BCH364C  Systems Biology & Bioinformatics (course # 55425 / 55330)
Spring 2023  Tue/Thu 11 – 12:30 PM  WEL 2.110
Instructor: Prof. Edward Marcotte  marcotte@utexas.edu
Zoom office hours: Mon 4 – 5

TA:  Matt McGuffie  mmcguffie@utexas.edu
Coding/problem set help hours:
Wed 3 – 4/Thu 12:30 – 1:30 in MBB 1.448BA
or by appointment on zoom

After hours Q/A, discussion:  Canvas

The class zoom channel will be posted on Canvas.
It will be the same zoom for class and office hours.

Probably the most important slide today!

Course web page:
http://www.marcottelab.org/
index.php/BCH394P_BCH364C_2023

This is a graduate student class!
It is open to a small # of upper division undergrads in natural sciences and engineering.

UG prerequisites:  Biochemistry 339F with a grade of at least B; Computer Science 303E and Statistics and Data Sciences 328M (or Statistics and Scientific Computation 318M, 328M) with a grade of at least C-; and consent of the instructor.
An introduction to systems biology and bioinformatics, emphasizing quantitative analysis of high-throughput biological data, and covering typical data, data analysis, and computer algorithms.

Topics will include introductory probability and statistics, basics of Python programming, protein and nucleic acid sequence analysis, genome sequencing and assembly, proteomics, synthetic biology, analysis of large-scale gene expression data, data clustering, biological pattern recognition, and gene and protein networks.

Note: it’s NOT a course on practical sequence analysis or using web-based tools. We’ll use these occasionally, but the focus will be on learning the underlying algorithms, exploratory data analyses, and their applications, esp. in high-throughput biology.

By the end of the course, you’ll know the fundamentals of important algorithms in bioinformatics and systems biology, be able to design and run computational studies in biology, and have performed an element of original computational biology research.
Books

Most of the lectures will be from research articles and slides. For sequence analysis, there will be an Optional text:

*Biological sequence analysis*, Durbin, Eddy, Krogh, Mitchison, Cambridge Univ. Press (available from Amazon, used & ebook)

For biologists rusty on their stats, *The Cartoon Guide to Statistics* (Gonick/Smith) is very good (really!).

We will also be learning intro Python programming. The course web site lists some recommendations to help you out, such as the free web course *Practical Python Programming* [https://dabeaz-course.github.io/practical-python/](https://dabeaz-course.github.io/practical-python/)

Important: There are bi-weekly coding/problem set help sessions. Plan to attend at least one per week!

Grading

No exams. Grades will be based on:

- **Online programming homework**
  (10 points each and counting 30% of the final grade)
- **3 problem sets**
  (15 points each and counting 45% of the final grade)
- **A course project** that you will develop over the semester & present in the last 3 days of class (25% of final grade)

The course project will consist of a research project on a bioinformatics topic chosen by the student (with approval by the instructor) containing an element of independent computational biology research (e.g. calculation, programming, database analysis, etc.) turned in as a web URL (20%) and presented in class (5%).

The project will be emailed as a web URL to the TA & I, developed through the semester and finished by 10 PM, April 12, 2023. The last few classes will be spent presenting your projects.
Late policy

• All projects and homework will be turned in electronically and time-stamped.

• No makeup work will be given.

• Instead, all students have 5 days of free “late time”. This is for the entire semester, NOT per project, and counting weekends/holidays just like any other day.
  • For projects turned in late, days will be deducted from the 5 day total (or what remains of it) by the # of days late.
  • Deductions are in 1 day increments, rounding up e.g. 10 minutes late = 1 day deducted.
  • Once the 5 days are used up, assignments will be penalized 10% / day late (rounding up), e.g., a 50 point assignment turned in 1 ½ days late would be penalized 20%, or 10 points.

Online homework will be via Rosalind: http://rosalind.info/faq/

Enroll specifically for BCH394P/364C at: https://rosalind.info/classes/enroll/fb013a1910/

The first homework will be due (in Rosalind) by 10 PM, Jan 18
Rosalind uses Python version 2, but we'll use version 3.

Rosalind uses the "vanilla" installation of Python. You're welcome to do it this way, but I recommend Anaconda/Jupyter as a nicer option.
Installing Anaconda/Jupyter

My recommendation for a good, all-round Python installation is **Anaconda**, available free to individuals here: **https://www.anaconda.com/products/individual**

***Get the latest Python 3 version***
(currently 3.9 in Anaconda, but any version >3 is probably fine)

Anaconda is a general management system for the various Python libraries and packages you might need, with >7,500 data science, visualization, and machine learning packages.

Anaconda also provides multiple Python interfaces. For this course, I recommend using **Jupyter Notebook**, which can be launched directly from the main Anaconda navigation window.

Jupyter is an interactive Python interface that shows your code & its output in successive entries in a shareable, archivable notebook viewable in any web browser, e.g.

Just type your command or code block & press “Shift-Enter” (or any of the various alternatives in the “cell” pulldown menu)

It’s widely used in bioinformatics and data visualization.
If you’re feeling restless/adventurous...

Back to Rosalind, for those of you that are a bit more advanced:

...there are quite a few good bioinformatics problems in the archives.
Expectations on working together

Students are welcome to discuss ideas and problems with each other, but all programs, Rosalind homework, problem sets, and written solutions should be performed independently, except the final presentation.

tl;dr: study/discuss together
do your own programming/writing/project
collaborate on the final presentation

What is Academic Dishonesty?

In promoting a high standard of academic integrity, the University broadly defines academic dishonesty—basically, all conduct that violates this standard, including any act designed to give an unfair or undeserved academic advantage, such as:

- Cheating
- Plagiarism
- Unauthorized Collaboration / Collusion
- Falsifying Academic Records
- Misrepresenting Facts (e.g., providing false information to postpone an exam, obtain an extended deadline for an assignment, or even gain an unearned financial benefit)
- Any other acts (or attempted acts) that violate the basic standard of academic integrity (e.g., multiple submissions—submitting essentially the same written assignment for two courses without authorization to do so)

https://deanofstudents.utexas.edu/conduct/academicintegrity.php
• By submitting as your own work any unattributed material that you obtained from other sources, you have committed plagiarism.

• Copying homework solutions from other students or internet sources (e.g. CourseHero) is cheating, collusion, and/or plagiarism.

• Software and computer code are legally considered in the same framework as other written works. Copying code directly without attribution is plagiarism.

• Any materials found online (e.g. CourseHero) that are associated with you, or any suspected unauthorized sharing of materials, will be reported to Student Conduct and Academic Integrity in the Office of the Dean of Students. These reports can result in sanctions, including failure in the course.

See the university’s official policy on plagiarism here: https://catalog.utexas.edu/general-information/appendices/appendix-c/student-discipline-and-conduct/
• You can use the internet to get ideas, programming suggestions and syntax, but downloading completed answers to assigned questions and submitting these as your own work is cheating/plagiarism.

• Copying entire programs verbatim from marked repositories offering Rosalind homework solutions is cheating and plagiarism.
Later in the semester, we’ll try co-programming with AI using chatGPT, where the goal is to make the computer write the code for you.

But there’s always an exception!

Why are we here? (practically, not existentially)
Our current-ish knowledge of human metabolism...

- Total number of reactions: 7,440
- Total number of metabolites: 5,063
- Number of unique metabolites: 2,626
- Number of metabolites in extracellular space: 642
- Number of metabolites in cytoplasm: 1,878
- Number of metabolites in mitochondrion: 754
- Number of metabolites in nucleus: 165
- Number of metabolites in endoplasmic reticulum: 570
- Number of metabolites in peroxisome: 435
- Number of metabolites in lysosome: 302
- Number of metabolites in Golgi apparatus: 317
- Number of transcripts: 2,194
- Number of unique genes: 1,789

Updated in Metabolomics 2016 12:109
Pales beside the phenomenal explosion of DNA sequencing:

Here are the latest statistics...

December 2022: 1.6 trillion bp Genbank + 19 trillion bp DNA whole genome shotgun sequencing

Which basically means GenBank is falling behind more every year!

Why are we here? We have no choice!

- Biologists are faced with a staggering deluge of data, growing exponentially

- Bioinformatics/comp bio tools and approaches help us understand these data and work productively, and to build increasingly powerful models of biological systems

- We’ll learn important basic concepts in this field and get exposed to key technologies driving the field
Specifically...

We’ll cover the following topics, approximately in this order:

**BASICS OF PROGRAMMING**
- Introduction to Rosalind
- A Python programming primer for non-programmers
- Rosalind help & programming Q/A

**BIOLOGICAL SEQUENCE ANALYSIS**
- Substitution matrices (BLOSSUM, PAM) & sequence alignment
- Protein and nucleic acid sequence alignments, dynamic programming
- Sequence profiles
- BLAST! (the algorithm)
- Biological databases
- Markov processes and Hidden Markov Models

**GENOMES, PROTEOMES, & "BIG BIOLOGY"**
- Gene finding algorithms
- Genome assembly & how the human genome was sequenced
- An introduction to large gene expression data sets
- Promoter and motif finding, Gibbs sampling
- Clustering algorithms, hierarchical, k-means, self-organizing maps, force-directed maps
- Classification algorithms
- Principal component analysis and data transformations

**NETWORK & SYNTHETIC BIOLOGY**
- Biological networks: metabolic, signaling, graphs, regulatory
- Deep homology and the evolution of traits
- Designing, simulating, and building gene circuits
- Genome design and synthesis
THE FINAL COURSE PROJECT IS DUE by 10 PM, April 12, 2023

The last 3 class days will be for presenting your projects

**Plus, expert guest lectures on:**

More advanced Python techniques  
NGS best practices  
Protein mass spectrometry / proteomics  
Protein 3D structural modeling