

FACULTY of ENGINEERING

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING,

BINF6112

COMPUTATIONAL BIOLOGY ENGINEERING DESIGN WORKSHOP

TERM 3, 2019

Course staff
Course Convener/Lecturer in Charge:
Mike Bain (MB) – m.bain@unsw.edu.au
School of Computer Science and Engineering: K17-401H, x56935
Lecturers
BG – Bruno Gaëta (bgaeta@unsw.edu.au)

Course details
This course is worth 6 units of credit and includes 4 hours of lectures and 1-2 hours of tutorial/lab time per week, although lectures are not held every week and some tutorial/lab slots are set aside for group presentations. Note that weeks without scheduled lectures or tutorial/lab slots are to allow time for group work on the project component of the course.

Course summary
This course examines various issues in the design and implementation of software systems for the analysis of large complex biological datasets and the application of mathematical and computational methods to problems in modern life science. Example domains include analysis of gene expression and function, networks of interaction, and systems modelling. See the course schedule for the syllabus.

Course aims and learning outcomes
To provide an overview of the field of computational methods of analysis and modelling in modern biology, together with its research and industrial context. By the end of this course you should be able to:
L1. Research and present on a range of open source software tools and techniques available to bioinformatics software developers and use at least one of these in your own software development projects
L2. Define and apply methods of clustering and classification in analysis of genome-scale data
L3. Express and apply key concepts from graph theory to model biological networks
L4. Research, use and explain selected engineering approaches to the representation, execution and evaluation of models of biological processes and their application in systems biology
L5. Demonstrate advanced practical design skills, particularly in software component integration and data management systems
L6. Successfully manage a project to completion in the light of complex requirements and tight budgets and schedules
L7. Communicate with stakeholders from a different background, especially domain experts in biology, through reports, presentations and documentation
L8. Explain ethical and legal issues associated with bioinformatics software systems especially concerning personal privacy, licensing and intellectual property

Assumed knowledge
Before commencing this course, students should:
- Be familiar with basic software engineering and development concepts and tools
- Be able to program in at least one programming language such as C, Java, Perl or Python

These are assumed to have been acquired in the courses COMP1531 or COMP2041, or equivalent.

Teaching strategies and rationale
- Lectures are structured to emphasise the computational biology analysis and modelling process. Lecture are related to relevant biology problems and context, then focus on computational concepts, algorithms and implementation issues.
- Lectures are supplemented by tutorial topics and computer lab work, and assignments that involve the implementation and application of software and interpretation of outputs.
- A range of project topics provide illustrations of applications in industry and current research.

UNSW graduate attributes especially developed in the course include:
- *The skills involved in scholarly enquiry* – students need to research, compare and evaluate different computational biology methods as part of coursework.

- *An in-depth engagement with the relevant disciplinary knowledge in its interdisciplinary context* – computational biology is presented in the context of its applications to biology, and of the computer science methods it draws on.
- *The capacity for analytical and critical thinking and for creative problem-solving* – tute/lab and project work requires students to solve a range of problems by choosing appropriate computational biology methods and applying them.
- *The ability to engage in independent and reflective learning* – the final examination especially requires students to reflect and provide a critical synthesis of the course contents.
- *A respect for ethical practice and social responsibility* – the course relates to issues of ethical considerations in data collection and analysis in computational biology applications.
- *The skills of effective communication* – written and oral communication is assessed principally through project reports and presentations, and the final examination.

Assessment

- Final exam 40%
  A formal exam in the exam period covering the lecture content of the course. (L2–4, L8).
- Project 40%
  Note: Although this project is done in a group, individual contributions will be peer-evaluated using the Moodle "team evaluation" module in order to allocate individual marks. (L1, L4–L6, L7–8).
    - Initial project presentation 10%
        - A 20 minute presentation by the team introducing their project and plan for development. Presentation should include background and motivation for the project, main requirements, and project management strategy. Mark for the presentation is given for the group as a whole.
    - Project design document (including requirements) and project plan 10%
        - A document listing requirements, user stories (including priority and scope, classified into epics), chosen technologies, major design decisions, test cases etc. It should also include a project plan outlining the proposed sprints (with planned deliverables for each sprint), team roles and responsibilities, meeting schedule, project management and version control software to be used
    - Final product 15%
        - A mark decided in consultation with the customer, reflecting the quality and usefulness of the final product.
    - Product documentation 5%
        - Documentation supplied with the final product, including both technical documentation and user manual. Assessed based on completeness and clarity of the document.
- Tute/lab solutions 10%
    - A series of two short self-directed data analysis and programming tasks where students work on some of the methods discussed in the lectures. (L2-4).
- Open source analytics and modelling presentations 10%
    - Group assignment to research into and present about a selected topic in biological system analysis and modelling. Marks will include a peer-assessed component as well as an individual presentation/contribution component. (L1, L6, L8).

Supplementary exams and assessment

A student will be offered a supplementary exam for the final exam only if they have missed the exam due to a well-documented medical reason. Students who sit a supplementary exam will have their marks calculated in the same way as other students. No supplementary exam will be awarded if the student has already sat the exam. Students must apply for special consideration in order to be considered for a supplementary exam.

Students with a final course mark of 47-49 may also be considered for supplementary assessment if it is allowed by UNSW policy. In this case, if the student passes the supplementary assessment they will receive a final mark of 50.

Course schedule

Lectures are held on Mondays 10-12 in Old Main Building 150 (K-K15-150) and on Tuesdays 10-12 in Ainsworth 202 (K-J17-202). The tutorial/lab slot is scheduled after the Tuesday lecture, on Tuesdays 2-4, in Webster 256 (K-G14-256). Since this course is framed as an engineering design workshop, a considerable amount of flexibility is built into the schedule, and all times are subject to change.

Week	**Lecture/Project**	**Tute/Lab/Project**	**Assessment due**
1: 16/9	Gene expression and functional genomics	Intro: confirmation of projects and groups	–
2: 23/9	Gene expression and functional genomics	Tute/lab questions 1	–
3: 30/9	Initial project presentations (if > 4 groups)	Initial project presentations	Tute/lab solutions 1
4: 7/10	Biological network analysis *(PUBLIC HOLIDAY on Monday)*	Tute/lab questions 2	Design and project plan
5: 14/10	Flexibility Week *(may use some time from another week in case of public holidays falling on a lecture/tutorial slot)*	–	Tute/lab solutions 2
6: 21/10	Systems biology modelling	Sprint review 1 - demo your prototype	–
7: 28/10	Systems biology modelling	Open source analytics and modelling (presentations)	–
8: 4/11	Sprint review 2 - demo your product (if > 4 groups)	Sprint review 2 - demo your product	–
9: 11/11	Ontologies and data integration	Open source analytics and modelling (presentations)	–
10: 18/11	Final demo and debrief (if > 4 groups)	Final product demo and debrief	Product documentation

Resources for students

- There is no required textbook for this course. However, the following books can be recommended as sources of introductory or deeper background reading and for further study on some of the course topics:
  "Introduction to Systems Biology – Design Principles of Biological Circuits (2nd Edition)" by Uri Alon, CRC Press, 2019.
  "Systems Biology – A Textbook (2nd Edition)" by Edda Klipp *et al*., Wiley-VCH, 2016.
  "Networks (2nd Edition)" by Mark Newman, Oxford University Press, 2018.
  "Nonlinear Dynamics and Chaos with Applications to Physics, Biology, Chemistry and Engineering (2nd Edition)" by Steven Strogatz, CRC Press, 2015.
  "Computational Modeling of Gene Regulatory Networks – A Primer" by Hamid Bolouri, Imperial College Press, 2008.

Lecture slides, discussion forums, announcements and specifications for assessed material will be made available on the course Moodle website (accessible through myUnsw)

Course evaluation and development

This will be the first time this course has run. Course content and delivery will be evaluated through the online MyExperience process at the end of the term. Individual lecturers may also distribute surveys on their own teaching. Feedback from these surveys is taken seriously and you are encouraged to respond.

Student Conduct

The Student Code of Conduct (Information at https://student.unsw.edu.au/conduct, Policy at https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf) sets out what the University expects from students as members of the UNSW community. As well as the learning, teaching and research environment, the University aims to provide an environment that enables students to achieve their full potential and to provide an experience consistent with the University's values and guiding principles. A condition of enrolment is that students *inform themselves* of the University's rules and policies affecting them, and conduct themselves accordingly.
In particular, students have the responsibility to observe standards of equity and respect in dealing with every member of the University community. This applies to all activities on UNSW premises and all external activities related to study and research. This includes behaviour in person as well as behaviour on social media, for example Facebook groups set up for the purpose of discussing UNSW courses or course work. Behaviour that is considered in breach of the Student Code Policy as discriminatory, sexually inappropriate, bullying, harassing, invading another's privacy or causing any person to fear for their personal safety is serious misconduct and can lead to severe penalties, including suspension or exclusion from UNSW.
If you have any concerns, you may raise them with your lecturer, or approach the School Ethics Officer (ethics-officer@cse.unsw.edu.au), Grievance Officer (grievance-officer@cse.unsw.edu.au), or one of the student representatives.
Plagiarism is defined as using the words or ideas of others and presenting them as your own. UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several on-line sources to help you understand what plagiarism is and how it is dealt with at UNSW:

Plagiarism and Academic Integrity: https://student.unsw.edu.au/plagiarism

UNSW Plagiarism Procedure: https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism. In particular, you are also responsible that your assignment files are not accessible by anyone but you by setting the correct permissions in your CSE directory and code repository, if using. Note also that plagiarism includes paying or asking another person to do a piece of work for you and then submitting it as your own work.

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.

If you haven't done so yet, please take the time to read the full text of UNSW's policy regarding academic honesty and plagiarism (https://student.unsw.edu.au/plagiarism)
The pages below describe the policies and procedures in more detail:

Student Code Policy:
https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf

Student Misconduct Procedure:
https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf

Plagiarism Policy Statement:
https://www.gs.unsw.edu.au/policy/documents/plagiarismpolicy.pdf

Plagiarism Procedure:
https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf

You should also read the following page, which describes your rights and responsibilities in the CSE context:
https://www.engineering.unsw.edu.au/computer-science-engineering/about-us/organisational-structure/student-services/policies/essential-advice-for-cse-students