Course Outline

Contents

- Course Details
- Course Summary
- Assumed Knowledge
- Student Learning Outcomes
- Teaching Strategies
- Teaching Rationale
- Student Conduct
- Assessment
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Details

Course Code	BINF6112
Course Title	Computational Biology Engineering Design Workshop
Convenor	Mike Bain (/users/z9400013) , Bruno Gaeta (/users/8670378)
Admin	Mike Bain (/users/z9400013)
Classes	*Lectures:* Wed 11:00 – 13:00 Online Thu 11:00 – 13:00 Online *Tutorials:* Fri 12:00-14:00 Online ***Note: lectures and tutorials will not be held every week to allow time for project work.*** Timetable for all classes (/BINF6112/20T23/timetable)
Consultations	By appointment
Units of Credit	6
Course Website	https://moodle.telt.unsw.edu.au/course/view.php?id... (https://moodle.telt.unsw.edu.au/course/view.php?id=53586) Note that webcms3 is used only for the course outline
Handbook Entry	http://www.handbook.unsw.edu.au/undergraduate/courses/current/BINF6112.html (http://www.handbook.unsw.edu.au/undergraduate/courses/current/BINF6112.html)

Course Summary

This course examines various issues in the design and implementation of software systems for the analysis and modelling of large complex biological datasets. The course also involves a guided team project for the design and implementation of a complex system bringing together a variety of tools and methods for analysing biological data. Methodologies for requirement gathering, system design, project management and documentation will be applied. The project work will be complemented by lectures on mathematical and computational methods to problems in modern life science.

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 including C, Java, Perl or Python

These are assumed to have been acquired in the courses COMP1531 or COMP2041 or COMP9021

- Be familiar with the fundamentals of bioinformatics applications as covered in the course BINF3010/9010

Student Learning Outcomes

This course aims to provide an overview of the field of computational methods of analysis and modelling in modern biology, together with its research and industrial context.

After completing this course, students will:

LO1. 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

LO2. Define and apply methods of clustering and classification in analysis of genome-scale data

LO3. Express and apply key concepts from graph theory to model biological networks

LO4. Research, use and explain selected engineering approaches to the representation, execution and evaluation of models of biological processes and their application in systems biology

LO5. Demonstrate advanced practical design skills, particularly in software component integration and data management systems

LO6. Successfully manage a project to completion in the light of complex requirements and tight budgets and schedules

LO7. Communicate with stakeholders from a different background, especially domain experts in biology, through reports, presentations and documentation

LO8. Explain ethical and legal issues associated with bioinformatics software systems especially concerning personal privacy, licensing and intellectual property

This course contributes to the development of the following graduate capabilities:

Graduate Capability	Acquired in
Scholars capable of independent and collaborative enquiry, rigorous in their analysis, critique and reflection, and able to innovate by applying their knowledge and skills to the solution of novel as well as routine problems	LO1-4.

Entrepreneurial leaders capable of initiating and embracing innovation and change, as well as engaging and enabling others to contribute to change	LO4-6.
Professionals capable of ethical, self- directed practice and independent lifelong learning	LO4-8.
Global citizens who are culturally adept and capable of respecting diversity and acting in a socially just and responsible way	LO5-8.

Teaching Strategies

The major project enables the students to work as a team to apply the lecture/tutorial content and design and implement a complex genome informatics management and analysis system, including project management, stakeholder presentations and documentation

The lectures introduce major concepts of algorithms used in the analysis of transcriptomics and systems biology data. Lectures are supplemented by group presentations on development and analytical environments for bioinformatics and tutorial work.

Online delivery

In 2020 the plan is to deliver the course online, although for the project component of the course, the option is available to run some face to face meetings for student presentations if enough students request it. In this eventuality your lecturer will contact you with information regarding on campus attendance and room locations. Lectures and tutorials will be delivered using Microsoft Teams or a similar delivery platform, as will project presentations. Lectures will be recorded but we encourage you to attend the presentation online so you can ask questions. For the team project, it is expected that you will be communicating online with your team members and your “customer”. Although this is not ideal, it provides useful training as collaborative software development is increasingly done remotely. Students who are overseas, or unable to attend campus will be able to complete this course in online mode including all classes and assessments. For face-to-face classes, we will be strictly following all government and UNSW guidelines on social distancing to ensure it is safe for you to attend classes. In the event of a serious outbreak of COVID-19 in Sydney, all classes may move to online mode. Please refer to https://www.covid-19.unsw.edu.au/safe-return-campus-faqs (https://www.covid-19.unsw.edu.au/safe-return-campus-faqs) and https://edtech.eng.unsw.edu.au/c19mess/comms.html (https://edtech.eng.unsw.edu.au/c19mess/comms.html) for further information.

Teaching Rationale

This is a practical course that requires students to learn by doing. Students undertake a group project with guidance from a mentor with set milestones throughout the session, including presentations, software demonstrations, and submission of reports and documentation. A series of lectures on computational biology algorithms complement the project work.

Student Conduct

The **Student Code of Conduct** (Information (https://student.unsw.edu.au/conduct) , Policy (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 (mailto:ethics-officer@cse.unsw.edu.au) , Grievance Officer (mailto:grievance-officer@cse.unsw.edu.au) , or one of the student representatives.

Plagiarism is defined as (https://student.unsw.edu.au/plagiarism) 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:

- Essential Advice for CSE Students (https://www.engineering.unsw.edu.au/computer-science-engineering/about-us/organisational-structure/student-services/policies/essential-advice-for-cse-students)

Assessment

Item	Topics	Due	Marks	Contributes to

Project	*Project plan 10% - due week 3* The initial project plan includes a group presentation (online) on which feedback will be provided, and a document which will be marked *Initial project presentation* 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. *Project design document (including requirements) and project plan* 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 20% - due week 10* A mark decided in consultation with the customer, reflecting the quality and usefulness of the final product. Although this is group work, individual contributions will be peer-evaluated using the moodle “team evaluation” module in order to allocate individual marks. *Product documentation 5% - due week 10* Documentation supplied with the final product, including both technical documentation and user manual. Assessed based on completeness and clarity of the document. *Ongoing project management 5% - assessed over the course of the project* A mark based on the ongoing management of the project over its course as shown in sprint reviews and product demo. Includes proper use of the tracking tool and sprint review presentations.	Weeks 3,10	40%	LO1, LO5-8
Tute/lab solutions	A series of two short self-directed data analysis and programming tasks where students work on some of the methods discussed in the lectures.	Weeks 4, 8	10%	LO1-3
Open source analytics and modelling presentations	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.	Week 9	10%	LO1
Final Exam	An exam in the exam period covering the lecture and tute/lab content of the course	Exam period	40%	LO1-4

Supplementary assessment policy

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 course mark in the range 45-49 will be assessed on a case-by-case basis.

Course Schedule

Week	Starting	Lec 1 (2hrs)	Lec 2 (2hrs)	Tute (2 hrs)	Assessment
1	Sep 14	Gene expression and functional genomics	Gene expression and functional genomics	**Project intro**	
2	Sep 21	Gene expression and functional genomics	Gene expression and functional genomics	Tute/lab (functional genomics)	
3	Sep 28	–	**Project ppts (if too many projects for a 2 hour session)**	**Project ppts**	Project plan 10%
4	Oct 5	Biological network analysis	Biological network analysis	–	Tute/lab solutions (1) 5%
5	Oct 12	Systems biology modelling	Systems biology modelling	**Sprint review 1**	
6	Oct 19	*Flexibility week*			
7	Oct 26	Systems biology modelling	Systems biology modelling	Tute/lab (networks)	
8	Nov 2	–	–	**Sprint review 2**	Tute/lab solutions (2) 5%
9	Nov 9	Ontologies and data integration	Ontologies and data integration	Open source analytics and modelling (presentations)	Presentations (10%)
10	Nov 10	–	**Final demos (if too many projects for a 2 hour session)**	**Final demos**	Final product 20% Product documentation 5%

Resources for Students

There are no required textbooks however recommended books for this course include:

- Bioinformatics Algorithms: An Active Learning Approach (Volumes I and II) Compeau and Pevzner, Active Learning Press (2015)
- Systems Biology – A Textbook. Klipp, Liebermeister, Wierling and Kowald, Wiley-VCH (2016)
- An Introduction to Systems Biology: Design Principles of Biological Circuits (2nd Edn.). Alon, Chapman and Hall/CRC (2019)

Lecture slides, discussion forums, announcements and assignment specs will be made available on the course moodle website (accessible through my.unsw)

Course Evaluation and Development

This course will be evaluated through the online MyExperience process at the end of session. Individual lecturers may also distribute surveys on their own teaching. Feedback from these surveys is taken seriously and you are encouraged to respond.

Resource created 9 months ago (Monday 07 September 2020, 08:37:21 AM), last modified 9 months ago (Sunday 20 September 2020, 09:10:01 PM).

Comments

(/BINF6112/20T3/forums/search?forum_choice=resource/52338) (/BINF6112/20T3/forums/resource/52338)

There are no comments yet.