Course Outline

Contents

- Course Details
- Course Summary
- Course Timetable
- Course Aims
- Student Learning Outcomes
- Assumed Knowledge
- Teaching Strategies
- Assessment
- Academic Honesty and Plagiarism
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Details

Course Code	COMP3141
Course Title	Software System Design and Implementation
Units of Credit	6
Course Website	http://cse.unsw.edu.au/~cs3141 (http://cse.unsw.edu.au/~cs3141)
Handbook Entry	http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP3141.html (http://www.handbook.unsw.edu.au/undergraduate/courses/current/COMP3141.html)

Course Summary

This course presents semi-formal and formal methods for the design and implementation phases of software system development. It introduces approaches to testing informed by formal designs, and it discusses trade-offs between static and dynamic approaches to improving software correctness. Throughout the course, the discussed methods are supported by software tools that assist in managing design, implementation, and testing. The course content is illustrated by case studies and practical exercises. Central topics are the use of logical properties and types to inform program design, implementation, validation, and verification. The course will introduces students to the strongly-typed Haskell programming language. No previous knowledge of Haskell is assumed.

Course Timetable

The course timetable is available here (/COMP3141/17s1/timetable) .

Course Aims

After completing this course, you should

- understand the how to use logical program properties to characterise aspects of the functional specification of programs.
- understand the difference and trade-offs between static methods (such as formal methods and type systems) and dynamic methods (such as testing) in assisting software design and implementation.
- understand the role of types in program design, implementation, validation, and verification. be able to use basic formal methods, strong type systems, and property-based testing to design and implement software while minimising software defects.
- be able to use a variety of tools based on formal specifications of logical properties.

The course exposes students to a mathematically founded approach to specifying and implementing software systems. It develops basic skills required to engineer software with high confidence in the correctness of the final product. The whole course encourages critical examination and analysis of existing solutions.

Assumed Knowledge

You need to have successfully completed the core programming, algorithm, and software development courses. You should be a confident coder and be prepared to study the concepts of a new programming language in directed self-study.

The prerequisites of COMP3141 are COMP1921 or COMP1927

Teaching Strategies

Lectures

The lectures will introduce you to new material, which is being reinforced and practised in weekly exercises and larger assignments. The course follows no particular textbook, but reading material covering specific topics will be identified throughout the course. Students are required to study reading material as advised during the lecture and/or on the course web page.

There are three hours of lectures each week.

Exercises

Weekly marked exercises start in Week 3. There will be 10 exercises, and these exercises will reinforce the material discussed in the lecture. Students have approximately one week to submit a solution. Submitting solutions to the exercises is compulsory, and solutions will be automatically marked. Solutions to exercises will not be accepted late. Automarking results are binding. Exercises will generally not be manually remarked. There are no extensions to exercises.

Assignments

There will be two practical assignments. They will be due approximately around Week 6 and Week 11. Students will have between one and two weeks to understand each individual assignment and to develop a solution.

Unless otherwise stated if you wish to submit an assignment late, you may do so, but a late penalty reducing the maximum available mark applies to every late assignment. The maximum available mark is reduced by 10% if the assignment is one day late, by 25% if it is 2 days late and by 50% if it is 3 days late. Assignments that are late 4 days or more will be awarded zero marks. So if your assignment is worth 88% and you submit it one day late you still get 88%, but if you submit it two days late you get 75%, three days late 50%, and four days late zero.

Assignment extensions are only awarded for serious and unforeseeable events. Having the flu for a few days, deleting your assignment by mistake, going on holiday, work commitments, and so on do not qualify. Therefore aim to complete your assignments well before the due date in case of last minute illness, and make regular backups of your work.

Assignments are being marked automatically. You need to make sure to follow the instructions closely. Failure to follow the details of the assignment specification is no reason for re-marking, even if small mistakes lead to a substantial loss of marks.

Assessment

Assessment consists of a practical component and a final examination. The break down of the practical component is as follows:

- Assignment 1: 20 marks
- Assignment 2: 20 marks
- Weekly Exercises: 60 marks

Both the practical component and the final examination are out of 100. The final grade for the course is determined by the harmonic mean between the practical component and the final examination.

Assignment and exercise work

Assignments and exercises are an important part of the course. They are an essential way of learning the practical skills you need to acquire. Any plagiarism in assignments or exercises will be severely punished and may result in an automatic Fail for the whole course. Read the plagiarism warning below for more details.

For each assignment, you will have approximately one week from release of the specification until the submission deadline. The specifications will be posted on the course web page.

Assignment work can be completed on the workstations at UNSW or on a computer at home. Your assignment must properly run on the computers at UNSW; so, test them at UNSW if you develop them at home. Unless otherwise stated, assignments must be submitted on-line from a school terminal using the give command. It is in your best interest to make regular backup copies of your work and (because of machine loads on deadline days, for example) to complete assignments well before their deadlines. Moreover, the electronic submission system give allows you to submit an assignment multiple times; only the last submission will be marked. We suggest that you submit a version once you have a partially complete solution and repeatedly submit whenever you improved your solution significantly. In particular, make sure that you submit your solution once you have completed the core component of each assignment. The core component of each assignment must be submitted to be able to pass the course.

Assignments

Each assignment is individual; i.e., no team work of any kind is permitted. Completing and submitting all assignments is compulsory; i.e., each assignment has a core component and you will not be permitted to pass the course unless you have made a reasonable effort to solve the core component. A "reasonable effort" means that there may be bugs in your solution, but you must submit an at least partially working piece of adequately structured code.

Exercises

There will be 10 weekly exercises (each valued at 6 marks). Each exercise is individual; i.e., no team work of any kind is permitted. You will not be allowed to pass the course unless you have submitted at least 8 of the weekly exercises (the submitted code can be only partially correct).

Final examination

The final exam is a three hour written exam. Requests for a supplementary exam will only be considered where students (a) have completed all other course components to a satisfactory standard, (b) have been absent from the final exam, (c) and have submitted a fully documented request for special consideration to NSQ within three working days of the final exam.

Examination hurdle

To achieve a passing final mark, a student needs to pass the exam. To pass the exam, a student needs to achieve 50% of overall marks awarded in the exam. Any student who fails the exam will automatically fail the entire course. No re-assessment will be awarded in this case.

Academic Honesty and Plagiarism

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, which can be found here (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)
- 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)
- Student Misconduct Procedure (https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf)

Course Schedule

Schedule of topics, organised by week ...

As in the following:

No schedule is available (yet)

Resources for Students

There is no textbook for the course. As tutorial and reference for Haskell, you can use either of these three books:

- Haskell Programming From First Principles by Christopher Allen and Julie Moronuki, pre-release.
- Real World Haskell by Bryan O'Sullivan, Don Stewart, and John Goerzen, O'Reilly Media.
- Learn You a Haskell for Great Good! by Miran Lipovača, No Starch Press.
  Both books are available as ebooks.

Further reading material will be announced in the lecture and/or on the course web page.

Getting Help

Questions regarding the course material, assignments, exercises, and general administrative questions should be asked on the course forum (accessible from the course web page), where answers benefit the whole class. Alternatively, approach the lecturer after class. To discuss matters concerning your personal performance, please send an email to the course account <cs3141@cse.unsw.edu.au>. For identification purposes, if you wish to send email concerning the course, you must: Send the mail from your CSE or UNSW student account (not from GMail, Yahoo, Bigpond or similar). Include your student id and your full name.

Course Evaluation and Development

This course is being continuously improved and we will conduct a survey at the end of session to obtain feedback on the quality of the various course components. Your participation in the survey will be greatly appreciated. Student feedback over the last years has generally been positive.

Resource created 5 years ago (Thursday 09 February 2017, 06:30:19 PM), last modified 5 years ago (Monday 13 February 2017, 02:54:07 PM).

Comments (/COMP3141/17s1/forums/search?forum_choice=resource/6221) (/COMP3141/17s1/forums/resource/6221)

There are no comments yet.