Australian National University | PROGRAMS AND COURSES

PROGRAMS AND COURSES / COURSES / COMP6710 / SECOND SEMESTER

CLASS NUMBER 5410 **TERM CODE** 3260

CLASS INFO

Unit Value

6 units

Mode of Delivery

In Person

COURSE CONVENER

AsPr Patrik Haslum

LECTURER

CLASS DATES

Class Start Date

25/07/2022

Class End Date

28/10/2022

Census Date

31/08/2022

Last Date to Enrol

CLASS NUMBER 5410 **TERM CODE** 3260

CLASS INFO

Unit Value

6 units

Mode of Delivery

In Person

COURSE CONVENER

AsPr Patrik Haslum

LECTURER

AsPr Patrik Haslum
Dr Giuseppe Maria Junior Barca

CLASS DATES

Class Start Date

25/07/2022

Class End Date

28/10/2022

Census Date

31/08/2022

Last Date to Enrol

01/08/2022

CLASS SUMMARY | BACK TO COURSE DETAILS

Structured Programming

A Postgraduate course offered by the **School of Computing**.

COMP6710

Overview Assessment Submission Contacts

This course introduces students to the fundamentals of software development with a substantial group software project at its center. Major foci are data structures, object oriented programming, and an introduction to software engineering. Students will extend their understanding of software productivity tools, using revision control for group work, and be introduced to test-driven development as an integral part of software construction.

Students will be introduced to an industrial strength object oriented programming language, extending their understanding of the imperative programming paradigm with a solid grounding in object oriented programming. Inheritance, polymorphism, and parametric types are taught, as well as concepts such as boxing and auto boxing. The important role of standard libraries and their collection types will be emphasized. GUI programming will be introduced.

The course includes a deeper treatment of data structures, using hash tables, trees and lists, which are used to provide concrete implementations of abstract library collection types. The theory of data structures and their time and space complexity will thus be tied to the practice of using standard collections such as those offered by object oriented languages.

The foundations of software engineering including: major development paradigms (such as big plan up front, agile, and formal methods) and risk are introduced.

Learning Outcomes

Upon successful completion, students will have the knowledge and skills to:

1. Apply fundamental programming concepts, using an object oriented programming language, to solve substantial problems
2. Understand basic types and the benefits of static typing for object oriented programs
3. Distinguish language definition from implementation, syntax and parsing from semantics and evaluation; understand how program state maps to memory (globals, local, heap) and the implications of heap reachability for memory management
4. Develop, understand, test, and evolve substantial programs using a modern IDE, and associated configuration tools; use programming approaches that avoid common coding errors; practice fundamental defensive programming; perform individual and team program reviews; use established design principles to organize a software system

5. Use, implement, and evaluate fundamental data structures and associated algorithms; create, implement, debug, and evaluate algorithms for solving substantial problems, including recursive, using divide-and-conquer and via decomposition; select and implement an abstract data type for a given problem
6. Perform analysis of simple algorithms; select and use appropriate algorithmic approaches to solve problems (brute-force, divide-and-conquer, recursive backtracking, heuristic)
7. Apply the even-driven programming paradigm to construct GUIs
8. Deliver and evaluate basic technical documents, presentations, and group interactions, using appropriate tools

Recommended Resources

Whether you are on campus or studying remotely, there are a variety of online platforms you will use to participate in your study program. These could include videos for lectures and other instruction, two-way video conferencing for interactive learning, email and other messaging tools for communication, interactive web apps for formative and collaborative activities, print and/or photo/scan for handwritten work and drawings, and home-based assessment.

ANU outlines recommended student system requirements to ensure you are able to participate fully in your learning. Other information is also available about the various Learning Platforms you may use.

Staff Feedback

Students will be given feedback in the following forms in this course:

- written comments
- verbal comments
- feedback to whole class, groups, individuals, focus group etc

Student Feedback

ANU is committed to the demonstration of educational excellence and regularly seeks feedback from students. Students are encouraged to offer feedback directly to their Course Convener or through their College and Course representatives (if applicable). Feedback can also be provided to Course Conveners and teachers via the Student Experience of Learning & Teaching (SELT) feedback program. SELT surveys are confidential and also provide the Colleges and ANU Executive with opportunities to recognise excellent teaching, and opportunities for improvement.

Other Information

https://comp.anu.edu.au/courses/comp1110/

Class Schedule

WEEK/SESSION	SUMMARY OF ACTIVITIES	ASSESSMENT
1	Lectures and lab (cf. course website)	
2	Lectures and lab (cf. course website)	
3	Lectures and lab (cf. course website)	
4	Lectures and lab (cf. course website)	assignments D1A, D2A (cf. course website)
5	Lectures and lab (cf. course website)	competency hurdle; assignment D2B (cf. course website)
6	Lectures and lab (cf. course website)	mid-sem exam (wk 6 or 7);
7	Lectures and lab (cf. course website)	mid-sem exam (wk 6 or 7); assignment D2C (cf. course website)
8	Lectures and lab (cf. course website)	assignment D2D, D2E (cf. course website)
9	Lectures and lab (cf. course website)	

WEEK/SESSION	SUMMARY OF ACTIVITIES	ASSESSMENT
10	Lectures (cf. course website)	
11	Lectures (cf. course website)	assignment D2F (cf. course website)
12	Lectures (cf. course website)	assignment D2G (cf. course website)

Assessment Summary

ASSESSMENT TASK	VALUE	LEARNING OUTCOMES
Individual assignment (A1) -- redeemable	5 %	1, 2, 3, 4
Group assignment (A2)	30 %	1, 2, 3, 4, 5, 6, 7, 8
Class engagement (CE) -- redeemable	5 %	4, 8
Lab test (LT) -- redeemable	5 %	1, 2, 3, 4
Mid-semester exam (M) -- redeemable	5 %	1, 2, 3, 4
Final exam (E)	50 %	1, 2, 3, 4, 5, 6
Basic competency test (BC)	0 %	1, 4

* If the Due Date and Return of Assessment date are blank, see the Assessment Tab for specific Assessment Task details

Policies

ANU has educational policies, procedures and guidelines , which are designed to ensure that staff and students are aware of the University's academic standards, and implement them. Students are expected to have read the Academic Integrity Rule before the commencement of their course. Other key policies and guidelines include:

- Academic Integrity Policy and Procedure
- Student Assessment (Coursework) Policy and Procedure
- Special Assessment Consideration Guideline and General Information
- Student Surveys and Evaluations
- Deferred Examinations
- Student Complaint Resolution Policy and Procedure
- Code of practice for teaching and learning

Responsible Officer: Registrar, Student Administration / **Page Contact:** Website Administrator / Frequently Asked Questions