Mechanical and Manufacturing Engineering

Course Outline

Semester 2 2018

MTRN3500

COMPUTING APPLICATIONS IN MECHATRONICS SYSTEMS

Contents

1. Staff contact details

Contact details and consultation times for course convenor

Name: Associate Professor Jay Katupitiya
Office location: Ainsworth 510E
Tel: (02) 9385 4096
Email: J.Katupitiya@unsw.edu.au
Moodle: https://moodle.telt.unsw.edu.au/login/index.php

For consultation please write to the above email address to make an appointment.

Please see the course Moodle.

2. Important links

- Moodle
- UNSW Mechanical and Manufacturing Engineering
- Course Outlines
- Student intranet
- UNSW Mechanical and Manufacturing Engineering Facebook
- UNSW Handbook

3. Course details

Credit Points

This is a 6 unit-of-credit (UoC) course, and involves 4 hours per week (h/w) of face-to-face contact.

The UNSW website states “The normal workload expectations of a student are approximately 25 hours per semester for each UoC, including class contact hours, other learning activities, preparation and time spent on all assessable work. Thus, for a full-time enrolled student, the normal workload, averaged across the 16 weeks of teaching, study and examination periods, is about 37.5 hours per week.”

This means that you should aim to spend about 9 h/w on this course. The additional time should be spent in making sure that you understand the lecture material, completing the set assignments, further reading, and revising for any examinations.

Contact hours

	Day	**Time**	**Location**
Lectures	Thursday	9 – 11 am	Colombo B
(Web)	Any	Any	Moodle

Labs	Monday	11am – 1pm	Willis Annexe 213
	Monday	1 pm – 3 pm	Willis Annexe 213
	Tuesday	2 pm – 4 pm	Willis Annexe 213
	Tuesday	6 pm – 8 pm	Willis Annexe 213
	Wednesday	11am – 1 pm	Willis Annexe 213
	Wednesday	1pm – 3 pm	Willis Annexe 213
	Wednesday	6 pm – 8 pm	Willis Annexe 213
	Thursday	11 am – 1 pm	Willis Annexe 213
	Thursday	1 pm – 3 pm	Willis Annexe 213
	Thursday	3 pm – 5 pm	Willis Annexe 213

Please refer to your class timetable for the learning activities you are enrolled in and attend only those classes.

Summary and Aims of the course

This course focuses on the continued learning of C++ with an emphasis on the application of C++ to real world programming tasks.

Aims of the Course

Description: The course is designed to complement the knowledge and skills gained by students in MTRN2500 so that they can develop application programs to deal with real world programming tasks. They will learn how to interface directly to sensors and actuators that are commonly used in Mechatronic Systems. The will then learn to put together different software modules designed by them or other individuals to form comprehensive software that are of commercial grade.

The courses in the Mechatronics discipline are built up on four different areas. They are; mechanical design, computing, electronics and microprocessors, and control systems. The latter three areas are interrelated, and this course forms a corner stone of the fundamental courses on which the Mechatronic Engineering course at UNSW is built upon. A high level of programming skills is necessary to develop customised interface routines to communicate with/control various elements of Mechatronic systems. This knowledge is essential in programming control systems and developing software modules for the interfacing of various hardware elements together to form complete Mechatronic Systems. As such the contributions from this course to the Mechatronic Engineering degree program are essential and vital.

Student learning outcomes

This course is designed to address the below learning outcomes and the corresponding Engineers Australia Stage 1 Competency Standards for Professional Engineers as shown. The full list of Stage 1 Competency Standards may be found in Appendix A.

After successfully completing this course, you should be able to:

Learning Outcome		EA Stage 1 Competencies
1.	Be well versed with structured and modular programming using C/C++ and to have appreciated the use of software to communicate with external devices.	PE1.1
2.	Be able to understand data structures, data transfer and transmission as well as inter-process communication.	PE1.1
3.	Be able to develop full software packages that are usable with commercial grade Mechatronic systems.	PE2.3

4. Teaching strategies

Teaching of this course is through web-based or face-to-face lectures and laboratory sessions. All laboratory work is individual work and attendance is preferred. The majority of the laboratory work involves some form of hardware. Initially, the hardware is predominantly sensors and various kinds of interface devices such as analogue and digital input/output devices. Towards the end of the course, actuators will be introduced.

The provision of the learning environment in the laboratory is to facilitate you to develop confidence in managing laboratory tasks as projects. The majority of the lab work involving actuators will be supervised by the demonstrators, professional officers and the academic in charge.

5. Course schedule

Week	Topic	Location	Suggested Readings
1	Introduction and Revision of C/C++	Colombo B	Moodle content
2	General Interfacing – Hardware descriptions	Colombo B	Moodle content
3	Programming Data Acquisition Systems	Colombo B	Moodle content
4	Serial/USB-Serial communication – Binary data sensors	Colombo B	Moodle content
5	Ethernet Communication – ASCII data sensors	Colombo B	Moodle content
6	Software architecture for Autonomous Systems	Colombo B	Moodle content
7	Shared Memory and Inter-process Communication	Colombo B	Moodle content
8	Process management	Colombo B	Moodle content
9	Selected vehicle models and their simulation	Colombo B	Moodle content
10	Implementing pure-pursuit for UGVs	Colombo B	Moodle content
11	Implementing attitude control for UAVs	Colombo B	Moodle content
12	Revision and Exam preparation	Colombo B	Moodle content

6. Assessment

Assessment overview

Assessment	Length	Weight	Learning outcomes assessed	Assessment criteria	Due date and submission requirements	Deadline for absolute fail	Marks returned
Programming assignment†	Completely operational software	20%	1 and 2	See assignment specification for exact marking criteria	Meeting with a demonstrator during week 6. (week finishes at 5 pm Friday)	5 pm Wednesday of Week 7	At the time of assessment
Programming assignment‡	Completely operational software	20%	3	See assignment specification for exact marking criteria	Meeting with a demonstrator during week 10.(week finishes at 5 pm Friday)	5 pm Wednesday of Week 11	At the time of assessment
Final exam	2 hours	60%	1, 2 and 3	All course content from weeks 1-12 inclusive.	Exam period, date TBC	N/A	Upon release of results

† The assignment specification will be available from week 2 onwards in Moodle.

‡ The assignment specification will be available from week 6 onwards in Moodle.

Assignments

Presentation

All programs must be explained fully to your demonstrator. A significant portion of the marks are for your knowledge demonstration during your meeting with the demonstrator.

At the end of the demonstrations, you must submit all your software in a zipped file form to Moodle submission site before midnight of the Friday of the week the assignment is assessed.

Submission

Work submitted late without an approved extension by the course coordinator or delegated authority is subject to a late penalty of 20 per cent (20%) of the maximum mark possible for that assessment item, per calendar day.

The late penalty is applied per calendar day (including weekends and public holidays) that the assessment is overdue. There is no pro-rata of the late penalty for submissions made part way through a day.

Work submitted after the ‘deadline for absolute fail’ is not accepted and a mark of zero will be awarded for that assessment item.

For some assessment items, a late penalty may not be appropriate. These are clearly indicated in the course outline, and such assessments receive a mark of zero if not completed by the specified date. Examples include:

a. Weekly online tests or laboratory work worth a small proportion of the subject mark, or
b. Online quizzes where answers are released to students on completion, or
c. Professional assessment tasks, where the intention is to create an authentic assessment that has an absolute submission date, or
d. Pass/Fail assessment tasks.

Marking

Marking guidelines for assignment submissions will be provided at the same time as assignment details to assist with meeting assessable requirements. Submissions will be marked according to the marking guidelines provided.

Examinations

You must be available for all tests and examinations. Final examinations for each course are held during the University examination periods, which are June for Semester 1 and November for Semester 2.

Provisional Examination timetables are generally published on myUNSW in May for Semester 1 and September for Semester 2

For further information on exams, please see the Exams section on the intranet.

Calculators

You will need to provide your own calculator, of a make and model approved by UNSW, for the examinations. The list of approved calculators is shown at student.unsw.edu.au/exam-approved-calculators-and-computers

It is your responsibility to ensure that your calculator is of an approved make and model, and to obtain an "Approved" sticker for it from the School Office or the Engineering Student Centre prior to the examination. Calculators not bearing an "Approved" sticker will not be allowed into the examination room.

Special consideration and supplementary assessment

For details of applying for special consideration and conditions for the award of supplementary assessment, see the information on UNSW's Special Consideration page.

7. Expected resources for students

Recommended Textbooks

- J.Katupitiya & K. Bentley, "Interfacing with C++", Springer 2006
- P.H. Winston, "On to C", Addison Wesley
- P.H. Winston, "On to C++", Addison Wesley

Additional Readings

The relevant chapters from the text book "Interfacing with C++" are available on Moodle Homepage of MTRN3500 together with a number of additional documents. Some materials from earlier years may also be available at Moodle's MTRN3500 Home page.

UNSW Library website: https://www.library.unsw.edu.au/
Moodle: https://moodle.telt.unsw.edu.au/login/index.php

8. Course evaluation and development

Feedback on the course is gathered periodically using various means, including the UNSW myExperience process, informal discussion in the final class for the course, and the School's Student/Staff meetings. Your feedback is taken seriously, and continual improvements are made to the course based, in part, on such feedback.

In this course, recent improvements resulting from student feedback include a set of tutorials to help you acquire programming skills based on lecture content.

9. Academic honesty and plagiarism

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW 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.*

Plagiarism is a type of intellectual theft. It can take many forms, from deliberate cheating to accidentally copying from a source without acknowledgement. UNSW has produced a website with a wealth of resources to support students to understand and avoid plagiarism: student.unsw.edu.au/plagiarism The Learning Centre assists students with understanding academic integrity and how not to plagiarise. They also hold workshops and can help students one-on-one.

You are also reminded that careful time management is an important part of study and one of the identified causes of plagiarism is poor time management. Students should allow sufficient time for research, drafting and the proper referencing of sources in preparing all assessment tasks.

If plagiarism is found in your work when you are in first year, your lecturer will offer you assistance to improve your academic skills. They may ask you to look at some online resources, attend the Learning Centre, or sometimes resubmit your work with the problem fixed. However more serious instances in first year, such as stealing another student's work or paying someone to do your work, may be investigated under the Student Misconduct Procedures.

Repeated plagiarism (even in first year), plagiarism after first year, or serious instances, may also be investigated under the Student Misconduct Procedures. The penalties under the procedures can include a reduction in marks, failing a course or for the most serious matters (like plagiarism in an honours thesis) even suspension from the university. The Student Misconduct Procedures are available here: www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf

Further information on School policy and procedures in the event of plagiarism is available on the intranet.

10. Administrative matters and links

All students are expected to read and be familiar with School guidelines and polices, available on the intranet. In particular, students should be familiar with the following:

- Attendance, Participation and Class Etiquette
- UNSW Email Address
- Computing Facilities

- Assessment Matters (including guidelines for assignments, exams and special consideration)
- Exams
- Approved Calculators
- Academic Honesty and Plagiarism
- Student Equity and Disabilities Unit
- Health and Safety
- Student Support Services

Appendix A: Engineers Australia (EA) Competencies

Stage 1 Competencies for Professional Engineers

	Program Intended Learning Outcomes
PE1: Knowledge and Skill Base	PE1.1 Comprehensive, theory-based understanding of underpinning fundamentals
	PE1.2 Conceptual understanding of underpinning maths, analysis, statistics, computing
	PE1.3 In-depth understanding of specialist bodies of knowledge
	PE1.4 Discernment of knowledge development and research directions
	PE1.5 Knowledge of engineering design practice
	PE1.6 Understanding of scope, principles, norms, accountabilities of sustainable engineering practice
PE2: Engineering Application Ability	PE2.1 Application of established engineering methods to complex problem solving
	PE2.2 Fluent application of engineering techniques, tools and resources
	PE2.3 Application of systematic engineering synthesis and design processes
	PE2.4 Application of systematic approaches to the conduct and management of engineering projects
PE3: Professional and Personal Attributes	PE3.1 Ethical conduct and professional accountability
	PE3.2 Effective oral and written communication (professional and lay domains)
	PE3.3 Creative, innovative and pro-active demeanour
	PE3.4 Professional use and management of information
	PE3.5 Orderly management of self, and professional conduct
	PE3.6 Effective team membership and team leadership