Mechanical and Manufacturing Engineering

Course Outline

Semester 2 2017

MTRN3500

Computing Applications in Mechatronic Systems

Contents

1. Staff contact details

Contact details and consultation times for course convenor

Name: Associate Professor Jay Katupitiya
Office location: Room ME510E, Ainsworth Building
Tel: (02) 9385 4096
Email: J.Katupitiya@unsw.edu.au

Consultation times: See News Forum after the start of the course.

Contact details and consultation times for additional lecturers/demonstrators/lab staff

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 four 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	Friday	9 am – 11 am	Colombo Theatre C
Demonstrations	Monday	12pm – 2 pm	MTRN213
	Monday	2 pm – 4 pm	MTRN213
	Tuesday	10 am – 12 pm	MTRN213
	Tuesday	12 pm – 2 pm	MTRN213
	Tuesday	2 pm – 4 pm	MTRN213
	Thursday	11 am – 1 pm	MTRN213
	Thursday	1 pm – 3 pm	MTRN213
	Friday	1 pm – 3 pm	MTRN213

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.

The course is designed to complement the knowledge and skills gained by students in MTRN2500 course 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. They 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 upon 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 cornerstone 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 absolutely 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 Comprehensive, theory-based understanding of underpinning fundamentals
2.	Be able to understand data structures, data transfer and transmission as well as inter-process communication.	PE1.1 Comprehensive, theory-based understanding of underpinning fundamentals
3.	Be able to develop full software packages that are usable with commercial grade Mechatronic systems.	PE2.3 Application of systematic engineering synthesis and design processes

4. Teaching strategies

Teaching of this course is through lectures and laboratory sessions. All laboratory work is individual work and attendance is preferred.

The provision of the learning environment in the laboratory is to facilitate you to develop confidence in managing laboratory tasks as projects. Demonstrators in the laboratories are there to provide you guidance and assistance in managing the laboratory tasks.

5. Course schedule

Topic	Date Fri (9am-11am)	Location	Lecture Content	Demo/ Lab	Suggested Readings
Revision of OOP	Week 1	Colombo C	Revision of Object Oriented Programming	None	Moodle lecture notes
General Interfacing	Week 2	Colombo C	Principles of input/output to hardware	None	Moodle lecture notes
Data acquisition systems	Week 3	Colombo C	Programming data acquisition systems	None	Moodle lecture notes
Serial Communication	Week 4	Colombo C	Reading sensors: ASCII data	None	Moodle lecture notes
Ethernet Communication	Week 5	Colombo C	Reading Sensors: Binary Data	None	Moodle lecture notes
Binary and ASCII Data	Week 6	Colombo C	Programming ASCII data sensors	None	Moodle lecture notes
File Streams	Week 7	Colombo C	Programming Binary data sensors	None	Moodle lecture notes
Reading Sensors	Week 8	Colombo C	Inter process communication and shared memory	None	Moodle lecture notes
Controlling actuators	Week 9	Colombo C	Process management	None	Moodle lecture notes
Inter-process communication	Week 10	Colombo C	UGV Software development project - programming	None	Moodle lecture notes
Multi-process management	Week 11	Colombo C	UGV Software development project - programming	None	Moodle lecture notes
Special topics & revision	Week 12	Colombo C	UGV Software development project - programming	None	Moodle lecture notes

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 Tuesday 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 Tuesday 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 final 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

Late submissions will be penalised 5 marks per calendar day (including weekends). An extension may only be granted in exceptional circumstances. Special consideration for assessment tasks must be processed through student.unsw.edu.au/special-consideration.

It is always worth submitting late assessment tasks when possible. Completion of the work, even late, may be taken into account in cases of special consideration.

Where there is no special consideration granted, the 'deadline for absolute fail' in the table above indicates the time after which a submitted assignment will not be marked, and will achieve a score of zero for the purpose of determining overall grade in the course.

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 School intranet, and the information on UNSW’s Special Consideration page.

7. Attendance

You are required to attend a minimum of 80% of all classes, including lectures, labs and seminars. It is possible to fail the course if your total absences equal to more than 20% of the required attendance. Please see the School intranet and the UNSW attendance page for more information.

8. 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 the 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

9. 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.

10. 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.

11. 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)
- 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