



Queensland, Australia

Course Code:	1004ENG
Course Name:	Computing & Programming with MATLAB
Semester:	Semester 1, 2016
Program:	Diploma of Engineering
Credit Points:	10
Course Coordinator:	Seyedali Mirjalili
Document modified:	24 Nov 2015 14:59:24

Teaching Team

Your lecturer/tutor can be contacted via the email system on the portal.	
Name	Email
Jason Ride	
Seyedali Mirjalili	ali.mirjalili@staff.griffithcollege.edu.au

Staff Consultation

Your lecturer/tutor is available each week for consultation outside of normal class times. Times that your lecturer/tutor will be available for consultation will be given in the first week of lectures. A list of times and rooms will be published on the Griffith College Portal under the "myTimetable" link.

Prerequisites

There are no prerequisites for this course

Brief Course Description

This course introduces students to MATLAB, a mathematical programming environment, and teaches generic programming techniques as well as MATLAB specific programming skills.

Rationale

The computer has become one of the most widely used tools in the modern world, and has profound impact on how engineering and science are practised today. Engineering also depends heavily on mathematical analysis and the MATLAB mathematical based computer program is used as the application tool for modeling and solving of engineering technical problems. The MATLAB mathematical programming environment has grown in importance to the engineering and scientific community and as such this foundation course is essential for students' wishing to pursue future studies where programming and computing become an integral part of the simulation/design methodology. Many of the skills learned in programming with MATLAB are also transferable to programming in other computer languages, the skills are also transferable into the workplace.

Aims

This course aims to introduce students to the engineering problem solving process in the context of high level structured computer programming. Throughout the course students will become familiar with both the hardware and software environments in a modern computing system, and be introduced to the fundamentals of computing that apply to all programming languages, and specifically to the MATLAB programming environment to solve various engineering problems. This course also aims to develop students' fundamental skills in problem conceptualisation, formulation, and solution in one of the most powerful and versatile programming environments - MATLAB.

Learning Outcomes

Upon successful completion of this course you will be able to:

1. Demonstrate a good understanding of the MATLAB programming environment.
2. Apply the programming theory to the MATLAB software environment.
3. Design algorithms and use MATLAB to solve engineering problems.
4. Understand the fundamentals and theory of computer programming.

Texts and Supporting Materials

Prescribed textbook:

Holly Moore, MATLAB for Engineers, Prentice Hall, 3E, 2011 (or later edition).

Recommended texts:

Stephen J. Chapman, MATLAB Programming for Engineers, 4E, Thomson, 2008.
David M. Smith, Engineering Computation with MATLAB, Addison Wesley, 2E, 2009.

SOFTWARE:

either: MATLAB (purchased from University Bookshops, purchased on-line from the co-op bookshop <http://www.coop.com.au/> or purchased on-line from mathworks.com)

or: Octave (download from <http://www.gnu.org/software/octave/>) (free, open-source) or: Scilab (download from <http://www.scilab.org/>) (free)

Octave & Scilab have varying levels of compatibility with Matlab and are useful in becoming familiar with Matlab.

Matlab programming is a skill that must be developed by practice. It is **strongly** advised that you have Matlab or one of the alternative software packages installed on your computer as soon as possible. Failure to practice Matlab programming will cause you to quickly fall behind in your studies.

Organisation and Teaching Strategies

The course will be taught using a combination of lectures, practical demonstration sessions, practical laboratory classes and self learning, designed to impart both theoretical understanding and hands-on experience of all topics covered. A total of 26 hours **lecture** (13 Å 2) will be presented during the semester, providing theory on computer systems, algorithm design, programming concepts and programming in MATLAB, with appropriate examples. A total of 12 hours **practical demonstration** (12 Å 1) will accompany the lectures for unpacking the theory before the laboratory sections. Practical reinforcement of the concepts will be undertaken in 36 hours **laboratory sessions** (12 Å 3 supervised practice), where you will be required to design and write MATLAB programs to solve a variety of real-world problems.

Class Contact Summary

Attendance:

The total contact hours for the course will consist of 26 hours of lectures (2 hours per week for 13 weeks), 12 hours practical demonstration (1 hour/week for weeks 1-12), and 36 hours of laboratory classes (3 hours/week for weeks 1-12). Your attendance in class will be marked twice during a four hour class. To receive full attendance, you must be present in the classroom on both occasions. Therefore, you are encouraged actively participate in all class sessions.

It is highly recommended that you attend all lectures, as these will provide the focus and theoretical foundations for the course. Two lecture sessions will include short in-class marked assessments.

Attendance at ALL the laboratory sessions in weeks 1-12 is also **COMPULSORY** recommended, as these will provide opportunity to develop the understanding of lecture material and the skills needed for the major assessment items in this course. A total of 8% of marks for this course are given for participation and completion of laboratory classes (1% for each laboratory completed, up to 8 out of the 12). Laboratory reports will be marked during the session. Because of the opportunities to complete eight assess labs there is no provision for marking deferred labs OR for marking labs missed due to late enrolment.

Participation in Class:

It is extremely important that you seek to actively participate in all classes throughout the semester.

Course Materials:

You are expected to acquire the required resources (textbook) for this course as the lectures follow the textbook throughout the semester.

Independent Learning:

Throughout this course you will be encouraged to take personal responsibility for managing your own learning and your own time. Each week throughout the semester you will be asked to prepare for classes by pre-reading the relevant chapters from the textbook. You are also expected to practice your programming skills in your private study time.

Program Progression

You are reminded that satisfactory Program Progression requires that attendance in classes is maintained at equal to or greater than 80% and that GPA is maintained at equal to or greater than 3.5 [please see Griffith College Policy Library - Program progression Policy - for more information].

Content Schedule

Weekly Teaching Schedule

Week	Topic	Activity	Readings
1	Introduction to computers, the history of computers, the speed of computers, the cost of computing etc. Introduction to the MATLAB Environment and the skills required to perform basic computations. Matlab M-files are introduced.	Lecture	Chapters 1 & 2
	Introduction to computers & Matlab Environment	Laboratory	Notes & Chapters 1 & 2
	Introduction to computers & Matlab Environment	Practical	Notes & Chapters 1 & 2
2	MATLAB Environment (cont)	Lecture	Chapter 1, 2
	Matlab Environment	Laboratory	Chapter 1, 2
	Matlab Environment	Practical	Chapter 1, 2
3	Algorithm Design and Programming Concepts – a standard problem-solving procedure is introduced for		

	program algorithm development; and an understanding is gained of why programming is widely used in engineering and science.	Lecture	Chapters 8 & 9 and Notes
	Algorithm Design & Programming Concepts	Laboratory	Chapters 8 & 9 and Notes
	Algorithm Design & Programming Concepts	Practical	Chapters 8 & 9 and Notes
4	Logical functions and control structures – describes logic functions and demonstrates how to create MATLAB code with control structures.	Lecture	Chapter 8 & 9
	Logical functions and control structures	Laboratory	Chapter 8 & 9
	Quiz on Topics: Introduction & Matlab Environment (Weeks 1 & 2)	Examination	During lectures
	Logical functions and control structures	Practical	Chapter 8 & 9
5	Build-in Matlab functions – details the wide variety of problems that can be solved with build-in MATLAB functions.	Lecture	Chapter 3
	Matlab Functions	Laboratory	Chapter 3
	Matlab Functions	Practical	Chapter 3
6	Manipulating MATLAB Matrices – demonstrates the power of formulating problems by using matrices in MATLAB and expanding on the techniques employed to define those matrices.	Lecture	Chapter 4
	Matrices	Laboratory	Chapter 4
	Quiz Practical programming quiz	Examination	During Lab session
	Matrices	Practical	Chapter 4
7	Plotting – learn to use MATLAB plotting skills to show engineering relationships and scientific findings	Lecture	Chapter 5
	Plotting	Laboratory	Chapter 5
	Plotting	Practical	Chapter 5
8	Plotting using workspace tools, editing plots using plot editors, editing plots using the mouse. Working with plot handles.	Lecture	Chapter 5 & Ch 14.2
	Plotting	Laboratory	Chapter 5
	Quiz on Topics: Algorithm Design / Pseudo Code, Flowcharts, Matlab logical functions and control structures	Examination	Weeks 3 & 4 Material During Lecture Session
	Plotting	Practical	Chapter 5
9	User-defined functions – describes how to create simple and complex user-defined functions and anonymous functions	Lecture	Chapter 6
	User-defined functions	Laboratory	Chapter 6
	User-defined functions	Practical	Chapter 6
10	Matrix Algebra Cell Arrays, Structures, Manipulating Strings (eg filenames)	Lecture	Chapter 10 & 11 & Notes
	String manipulation, using cell arrays, structures and matrix math	Laboratory	Chapter 10 & 11 & Notes
	Quiz Practical Programming Quiz	Examination	During Lab Session
	String manipulation, using cell arrays, structures and matrix math	Practical	Chapter 10 & 11 & Notes
11	User-controlled input/output – introduces functions that interact with the program users, including user-defined input, formatted output and graphic input techniques.	Lecture	Chapter 7
	Assignment Due	Class	Due Friday
	User controlled input & output	Laboratory	Chapter 7
	User controlled input & output	Practical	Chapter 7
12	Advanced Topics – Numerical computing, interpolation and curve fitting	Lecture	Chapter 13
	Advanced Problem Solving	Laboratory	Notes
	Advanced Problem Solving	Practical	Notes
13	Review key matlab functionality and core functional areas	Lecture	Chapter 1-11 & Notes
14	Final Examination	Examination	Date and time to be advised.

This section sets out the assessment requirements for this course.

Summary of Assessment

Item	Assessment Task	Weighting	Relevant Learning Outcomes	Due Date
1	Laboratories	8%	1,2,3,4	1-12
2	Quiz 1 - In Class	8%	1	4
3	Quiz 2 - Lab Quiz	8%	1,2	6
4	Quiz 3 - In Class	8%	3	8
5	Quiz 4 - Lab Quiz	8%	2,3,4	10
6	Assignment: Programming and written report	20%	1,2,3&4	11
7	Final Examination	40%	2,3&4	14

Assessment Details

Laboratories:

You will be awarded up to 1% of marks for the effort at each laboratory session up to a maximum of 8%. Laboratory marks are designed to encourage you to attend laboratory sessions and to develop and reinforce your understanding and application of programming concepts in practice. Your best eight Lab results will count towards your Lab total. Because of the opportunities to complete eight assessed labs there is no provision for marking deferred labs OR for marking labs missed due to late enrolment.

In Class Quizzes

The purpose of the quizzes is to motivate you and assess theoretical understanding of the conceptual material delivered in lectures, and developed through personal study and laboratory experience. The quizzes will be completed during the lecture in week 4 and 8. Attendance at these lectures is therefore compulsory. The quizzes will also provide you feedback on how you are progressing in the course.

Lab Quizzes

The purpose of the quizzes is to motivate you and assess your practical ability to write Matlab programs as solutions to given problems. The quizzes will be completed during the laboratory sessions in week 6 and 10. Attendance at these laboratory sessions is therefore compulsory. The quizzes will also provide you feedback on how you are progressing in the course.

Assignment:

The assignment consists of real engineering/science problems that you are required to solve using the knowledge and programming skills learnt from the course. A written report is required to be submitted with the programming codes. Details of these problems will be provided on an assignment sheet available from the course website:

Final Examination:

The purpose of the final examination is to examine your knowledge and skills acquired during the course. The examination will be held in the central examination period at the end of semester.

NOTE: Students must achieve an overall mark of 50% in the courses to be eligible for a pass mark for the course.

Submission and Return of Assessment Items

Normally you will be able to collect your assignments in class within fourteen [14] days of the due date for submission of the assignment.

Retention of Originals

You must be able to produce a copy of all work submitted if so requested. Copies should be retained until after the release of final results for the course.

Extensions

To apply for an extension of time for an assignment, you must submit an Application for Extension of Assignment form to your teacher at least 24 hours before the date the assignment is due. Grounds for extensions are usually: serious illness, accident, disability, bereavement or other compassionate circumstances and must be able to be substantiated with relevant documentation [e.g. Griffith College Medical Certificate]. Please refer to the Griffith College website - Policy Library - for guidelines regarding extensions and deferred assessment.

Assessment Feedback

Marks awarded for assessment items will also be available on the on-line grades system on the Student Website within fourteen [14] days of the due date.

Generic Skills

Griffith College aims to develop graduates who have an open and critical approach to learning and a capacity for lifelong learning. Through engagement in their studies, students are provided with opportunities to begin the development of these and other generic skills.

Studies in this course will give you opportunities to begin to develop the following skills:

Generic Skills	Taught	Practised	Assessed
Written Communication		Yes	
Oral Communication		Yes	
Information Literacy	Yes	Yes	Yes
Secondary Research		Yes	
Critical and Innovative Thinking		Yes	
Academic Integrity	Yes	Yes	Yes
Self Directed Learning		Yes	
Team Work		Yes	
Cultural Intelligence			
English Language Proficiency			

Additional Course Generic Skills

Additional Course Information

In addition to formal contact hours, you are provided with extra support through individual consultation with teaching staff, tutorials in English language, and self-access computer laboratories.

Teacher and Course Evaluations

Students enjoy the challenge of learning and working with advanced Engineering software. In response to student feedback to provide them with more time and support to complete course assessment items, teacher availability was increased so that discussions on assessment items such as assignments and labs could be facilitated.

Your feedback is respected and valued by your lecturers and tutors. You are encouraged to provide your thoughts on the course and teaching, both positive and critical, directly to your lecturer and tutor or by completing course and lecturer evaluations via Griffith College's online evaluation tool whenever these are available.

Academic Integrity

Griffith College is committed to maintaining high academic standards to protect the value of its qualifications. Academic integrity means acting with the values of honesty, trust, fairness, respect and responsibility in learning, teaching and research. It is important for students, teachers, researchers and all staff to act in an honest way, be responsible for their actions, and show fairness in every part of their work. Academic integrity is important for an individual's and the College's reputation.

All staff and students of the College are responsible for academic integrity. As a student, you are expected to conduct your studies honestly, ethically and in accordance with accepted standards of academic conduct. Any form of academic conduct that is contrary to these standards is considered a breach of academic integrity and is unacceptable.

Some students deliberately breach academic integrity standards with intent to deceive. This conscious, pre-meditated form of cheating is considered to be one of the most serious forms of fraudulent academic behaviour, for which the College has zero tolerance and for which penalties, including exclusion from the College, will be applied.

However, Griffith College also recognises many students breach academic integrity standards without intent to deceive. In these cases, students may be required to undertake additional educational activities to remediate their behaviour and may also be provided appropriate advice by academic staff.

As you undertake your studies at Griffith College, your lecturers, tutors and academic advisors will provide you with guidance to understand and maintain academic integrity; however, it is also your responsibility to seek out guidance if and when you are unsure about appropriate academic conduct.

Please ensure that you are familiar with the [Griffith College Academic Integrity Policy](#); this policy provides an overview of some of the behaviours that are considered breaches of academic integrity, as well as the penalties and processes involved when a breach is identified.

For further information please refer to the Academic Integrity Policy on the Griffith College website – Policy Library.

Risk Assessment Statement

Classroom & Computer Lab use only. No out of the ordinary risks are associated with this course.

Copyright © - Griffith College

Note: For all Diploma level programs, Griffith College acknowledges content derived from Griffith University.