433-330 Theory of Computation

Credit Points:	12.50
Level:	3 (Undergraduate)
Dates & Locations:	2009, This subject commences in the following study period/s: Semester 1, - Taught on campus.
Time Commitment:	Contact Hours: Twenty-four hours of lectures and approximately 11 hours of tutorials Total Time Commitment: Not available
Prerequisites:	433-253 Algorithms and Data Structures and 433-255 Logic and Computation
Corequisites:	None
Recommended Background Knowledge:	None
Non Allowed Subjects:	None
Core Participation Requirements:	For the purposes of considering request for Reasonable Adjustments under the Disability Standards for Education (Cwth 2005), and Student Support and Engagement Policy, academic requirements for this subject are articulated in the Subject Overview, Learning Outcomes, Assessment and Generic Skills sections of this entry. It is University policy to take all reasonable steps to minimise the impact of disability upon academic study, and reasonable adjustments will be made to enhance a student's participation in the University's programs. Students who feel their disability may impact on meeting the requirements of this subject are encouraged to discuss this matter with a Faculty Student Adviser and Student Equity and Disability Support: http://services.unimelb.edu.au/disability
Coordinator:	Assoc Prof Harald Christian Sondergaard
Subject Overview:	Topics covered include formal languages, grammars and recognisers; models of computation: finite state machines, pushdown automata, Turing machines; computability: the Church-Turing thesis, decidability, reducibility; complexity: the classes P and NP, NP-complete problems, space complexity; and other topics selected from information and coding theory, lambda calculus, recursion theory, approximation algorithms, probabilistic algorithms, cryptography, quantum computing.
Objectives:	On completion of this subject students should understand the essence of computing through simple models of computational devices; understand the limitations of computing, the relative power of formal languages and the inherent complexity of many computational problems of practical importance; be able to apply these models in practice to solving problems in diverse areas such as string searching, pattern matching, cryptography, and language design; be familiar with standard tools and notation for formal reasoning about machines and programs; and be able to improve reasoning and problem-solving skills.
Assessment:	Project work during semester, expected to take about 36 hours (24%); and a 3-hour end-of-semester written open-book examination (76%). To pass the subject, students must obtain at least 50% overall, 12/24 in project work, and 38/76 in the written examination.
Prescribed Texts:	None
Breadth Options:	This subject is not available as a breadth subject.
Fees Information:	Subject EFTSL, Level, Discipline & Census Date, http://enrolment.unimelb.edu.au/fees
Generic Skills:	On completion students should have an: # ability to apply knowledge of basic science and engineering fundamentals; and

	# ability to undertake problem identification, formulation and solution.
Notes:	Students enrolled in the BSc (pre-2008 BSc), BASc or a combined BSc course will receive science credit for the completion of this subject.
Related Course(s):	Bachelor of Engineering (Computer Engineering) Bachelor of Engineering (Electrical Engineering) Bachelor of Engineering (Mechatronics) and Bachelor of Computer Science Bachelor of Engineering (Software Engineering)
Related Majors/Minors/ Specialisations:	Computer Science Computer Science Major Logic and Philosophy of Science