

COMP90046 Constraint Programming

Credit Points:	12.50															
Level:	9 (Graduate/Postgraduate)															
Dates & Locations:	This subject is not offered in 2013.															
Time Commitment:	Contact Hours: 36 hours, comprising of two 1-hour lectures and one 1-hour workshop per week Total Time Commitment: 120 hours															
Prerequisites:	<p>One of the following:</p> <table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>COMP20003 Algorithms and Data Structures</td> <td>Not offered 2013</td> <td>12.50</td> </tr> <tr> <td>COMP20007 Design of Algorithms</td> <td>Not offered 2013</td> <td>12.50</td> </tr> <tr> <td>COMP90038 Algorithms and Complexity</td> <td>Not offered 2013</td> <td>12.50</td> </tr> <tr> <td>COMP20006 Programming the Machine</td> <td>Not offered 2013</td> <td>12.50</td> </tr> </tbody> </table>	Subject	Study Period Commencement:	Credit Points:	COMP20003 Algorithms and Data Structures	Not offered 2013	12.50	COMP20007 Design of Algorithms	Not offered 2013	12.50	COMP90038 Algorithms and Complexity	Not offered 2013	12.50	COMP20006 Programming the Machine	Not offered 2013	12.50
Subject	Study Period Commencement:	Credit Points:														
COMP20003 Algorithms and Data Structures	Not offered 2013	12.50														
COMP20007 Design of Algorithms	Not offered 2013	12.50														
COMP90038 Algorithms and Complexity	Not offered 2013	12.50														
COMP20006 Programming the Machine	Not offered 2013	12.50														
Corequisites:	None															
Recommended Background Knowledge:	None															
Non Allowed Subjects:	433-433 Constraint Programming 433-633 Constraint Programming 433-671 Constraint Programming															
Core Participation Requirements:	<p><p>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.</p> <p>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</p></p>															
Contact:	Associate Professor Tim Baldwin email: tbaldwin@unimelb.edu.au (mailto:tbaldwin@unimelb.edu.au)															
Subject Overview:	<p>Constraint programming is used to solve constrained satisfaction and optimization problems such as scheduling and allocation, which are of vital importance to modern business. Constraint programming allows these problems to be modeled at a high level, and allows the solving to take advantage of the structure of the problem.</p> <p>Topics covered will be:</p> <ul style="list-style-type: none"> # Modelling with constraints # Algorithms for manipulating constraints # Finite domain constraint solving # Global constraints # Programming search # Hybrid constraint solving 															

Objectives:	<p>On completion of this subject students should be able to:</p> <ul style="list-style-type: none"> # Model a complex constraint problem using a high level modeling language # Define and explore different search strategies for solving a problem # Explain how modelling interacts with the solving algorithms, and formulate models to take advantage of this # Use state of the art optimisation tools
Assessment:	<p>Three projects due in weeks 4, 8 and 12 (30%) One examination not exceeding 3 hours, held in the examination period (70%) Hurdle Requirement: Students must obtain a mark of at least 35/70 on the exam</p>
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:	<p>On completion of this subject students should be able to have the:</p> <ul style="list-style-type: none"> # Ability to undertake problem identification, formulation, and solution # Ability to utilise a systems approach to complex problems and to design and for operational performance # Ability to manage information and documentation # Capacity for creativity and innovation
Related Course(s):	<p>Master of Engineering in Distributed Computing Master of Information Technology Master of Information Technology Master of Information Technology Master of Philosophy - Engineering Master of Science (Computer Science) Ph.D.- Engineering</p>
Related Majors/Minors/ Specialisations:	<p>B-ENG Software Engineering stream Computer Science Master of Engineering (Software)</p>