

# COMP90046 Constraint Programming

<b>Credit Points:</b>	12.50						
<b>Level:</b>	9 (Graduate/Postgraduate)						
<b>Dates &amp; Locations:</b>	This subject is not offered in 2011.						
<b>Time Commitment:</b>	Contact Hours: 36 hours, made up of 24 one-hour lectures (two per week) and 12 one-hour workshops (one per week) Total Time Commitment: 120 hours						
<b>Prerequisites:</b>	<p>The prerequisites are:</p> <table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>COMP90048 Declarative Programming</td> <td>Not offered 2011</td> <td>12.50</td> </tr> </tbody> </table>	Subject	Study Period Commencement:	Credit Points:	COMP90048 Declarative Programming	Not offered 2011	12.50
Subject	Study Period Commencement:	Credit Points:					
COMP90048 Declarative Programming	Not offered 2011	12.50					
<b>Corequisites:</b>	None						
<b>Recommended Background Knowledge:</b>	None						
<b>Non Allowed Subjects:</b>	433-433 Constraint Programming 433-633 Constraint Programming 433-671 Constraint Programming						
<b>Core Participation Requirements:</b>	For the purposes of considering request for Reasonable Adjustments under the Disability Standards for Education (Cwth 2005), and Students Experiencing Academic Disadvantage Policy, academic requirements for this subject are articulated in the Subject Description, Subject Objectives, Generic Skills and Assessment Requirements of this entry. The University is dedicated to provide support to those with special requirements. Further details on the Disability support scheme can be found at the Disability Liaison Unit Website: <a href="http://www.services.unimelb.edu.au/disability/">http://www.services.unimelb.edu.au/disability/</a>						
<b>Contact:</b>	Associate Professor Tim Baldwin email: <a href="mailto:tbaldwin@unimelb.edu.au">tbaldwin@unimelb.edu.au</a> ( <a href="mailto:tbaldwin@unimelb.edu.au">mailto:tbaldwin@unimelb.edu.au</a> )						
<b>Subject Overview:</b>	<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"> <li>modelling with constraints</li> <li>algorithms for manipulating constraints</li> <li>finite domain constraint solving</li> <li>global constraints</li> <li>programming search</li> <li>hybrid constraint solving</li> </ul>						
<b>Objectives:</b>	<p>On completion of this subject students should be able to:</p> <ul style="list-style-type: none"> <li># Model a complex constraint problem using a high level modeling language</li> <li># Define and explore different search strategies for solving a problem</li> <li># Explain how modelling interacts with the solving algorithms, and formulate models to take advantage of this</li> <li># Use state of the art optimisation tools</li> </ul>						
<b>Assessment:</b>	Three projects due in weeks 4, 8 and 12 (30%), and an end-of-semester exam not exceeding 3 hours in duration (70%). To pass the subject, students must obtain a mark of at least 35/70 on the exam.						

<b>Prescribed Texts:</b>	None
<b>Breadth Options:</b>	This subject is not available as a breadth subject.
<b>Fees Information:</b>	Subject EFTSL, Level, Discipline & Census Date, <a href="http://enrolment.unimelb.edu.au/fees">http://enrolment.unimelb.edu.au/fees</a>
<b>Generic Skills:</b>	On completion of this subject students should be able to have the: <ul style="list-style-type: none"><li># Ability to undertake problem identification, formulation, and solution</li><li># Ability to utilise a systems approach to complex problems and to design and for operational performance</li><li># Ability to manage information and documentation</li><li># Capacity for creativity and innovation</li></ul>
<b>Related Majors/Minors/ Specialisations:</b>	B-ENG Software Engineering stream Master of Engineering (Software)