

# COMP90054 Software Agents

<b>Credit Points:</b>	12.50									
<b>Level:</b>	9 (Graduate/Postgraduate)									
<b>Dates &amp; Locations:</b>	This subject is not offered in 2013.									
<b>Time Commitment:</b>	Contact Hours: 36 hours, comprising of two 1-hour lectures and one 1-hour workshop per week Total Time Commitment: 120 hours									
<b>Prerequisites:</b>	<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>COMP90048 Declarative Programming</td> <td>Not offered 2013</td> <td>12.50</td> </tr> <tr> <td>COMP30020 Declarative Programming</td> <td>Not offered 2013</td> <td>12.50</td> </tr> </tbody> </table>	Subject	Study Period Commencement:	Credit Points:	COMP90048 Declarative Programming	Not offered 2013	12.50	COMP30020 Declarative Programming	Not offered 2013	12.50
Subject	Study Period Commencement:	Credit Points:								
COMP90048 Declarative Programming	Not offered 2013	12.50								
COMP30020 Declarative Programming	Not offered 2013	12.50								
<b>Corequisites:</b>	None									
<b>Recommended Background Knowledge:</b>	None									
<b>Non Allowed Subjects:</b>	433-481 Agent Programming Languages 433-482 Software Agents 433-681 Agent Programming Languages 433-682 Software Agents									
<b>Core Participation Requirements:</b>	<p>&lt;p&gt;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.&lt;/p&gt; &lt;p&gt;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: &lt;a href="http://services.unimelb.edu.au/disability"&gt;http://services.unimelb.edu.au/disability&lt;/a&gt;&lt;/p&gt;</p>									
<b>Contact:</b>	Dr Adrian Pearce email: <a href="mailto:adrianrp@unimelb.edu.au">adrianrp@unimelb.edu.au</a> (mailto:adrianrp@unimelb.edu.au)									
<b>Subject Overview:</b>	<p>Software agents are software systems which use automated reasoning algorithms operating on symbolic knowledge representations to determine how they should interact with their environment (the real world, human beings, or other agents) in order to achieve the goals set for them. Agent modelling and programming techniques are becoming increasingly popular for solving problems for robotic, artificially intelligent or internetworking processes.</p> <p>The subject covers the principles of agent-oriented modelling and the underlying automated reasoning and agent programming techniques involved. Topics covered include: agent modelling of goals, roles and organisations, planning, sensing and acting, introduction to modal logic and model theory, epistemic logic, theorem proving, dynamic logic, the situation calculus, logical regression, non-deterministic programming, concurrency and multi-agent programming. The subject focuses on the foundations that enable agents to reason autonomously about goals, perception, actions and the knowledge of other agents during collaborative task execution.</p>									
<b>Objectives:</b>	On completion of this subject students should be able to:									

	<ul style="list-style-type: none"> <li># Apply agent modelling techniques to analyse, design and implement a small agent-based system</li> <li># Perform logical analysis of distributed problems based on (at least one of) modal logic, dynamic logic, epistemic logic and model theory techniques</li> <li># Program effective agent reasoning strategies using an action language (such as the situation calculus) and high-level, non-deterministic programming principles</li> <li># Choose and apply effective concurrent programming and continuous planning techniques to non-trivial distributed problems</li> </ul>
<b>Assessment:</b>	Two 1750 word assignments, due in approximately weeks 6 and 11 (30% each) A 15 minute in-class oral presentation concerning assignment work (10%) One 2-hour end-of-semester open-book examination (30%)
<b>Prescribed Texts:</b>	Leon Sterling and Kuldar Taveter, The Art of Agent-Oriented Modelling, MIT Press, 2009.
<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>	<p>On completion of the subject the students should have the:</p> <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 operational performance</li> <li># Ability to manage information and documentation</li> <li># Capacity for creativity and innovation Ability to communicate effectively with both the engineering team and the community at large</li> </ul>
<b>Related Course(s):</b>	<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)  Master of Software Systems Engineering  Ph.D.- Engineering</p>
<b>Related Majors/Minors/Specialisations:</b>	<p>B-ENG Software Engineering stream  Master of Engineering (Software)</p>