COMP90054 Software Agents

<table>
<tr><td><b>Credit Points:</b></td><td>12.5</td></tr>
<tr><td><b>Level:</b></td><td>9 (Graduate/Postgraduate)</td></tr>
<tr><td><b>Dates &amp; Locations:</b></td><td>2015, Parkville
This subject commences in the following study period/s:
Semester 2, Parkville - Taught on campus.</td></tr>
<tr><td><b>Time Commitment:</b></td><td>Contact Hours: 36 hours, comprising of two 1-hour lectures and one 1-hour workshop per week
Total Time Commitment: 200 hours</td></tr>
<tr><td><b>Prerequisites:</b></td><td><b>One of the following:</b>
<table>
<tr><th>Subject</th><th>Study Period Commencement:</th><th>Credit Points:</th></tr>
<tr><td>COMP90048 Declarative Programming</td><td>Semester 2</td><td>12.50</td></tr>
<tr><td>COMP30020 Declarative Programming</td><td>Semester 2</td><td>12.50</td></tr>
</table></td></tr>
<tr><td><b>Corequisites:</b></td><td>None</td></tr>
<tr><td><b>Recommended Background Knowledge:</b></td><td>None</td></tr>
<tr><td><b>Non Allowed Subjects:</b></td><td>433-481 Agent Programming Languages
433-482 Software Agents
433-681 Agent Programming Languages
433-682 Software Agents</td></tr>
<tr><td><b>Core Participation Requirements:</b></td><td><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: <a href="http://services.unimelb.edu.au/disability">http://services.unimelb.edu.au/disability</a></p></td></tr>
<tr><td><b>Coordinator:</b></td><td>Assoc Prof Adrian Pearce</td></tr>
<tr><td><b>Contact:</b></td><td>email: <b>adrianrp@unimelb.edu.au (mailto:adrianrp@unimelb.edu.au)</b></td></tr>
<tr><td><b>Subject Overview:</b></td><td><b>AIMS</b>
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. The subject covers the principles of agent-oriented modelling and the underlying automated reasoning and planning techniques involved. 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. A key focus of this subject is automated planning for real-world problems such as robotics.
This subject is an elective subject in the Master of Information Technology, in particular for the Distributed Computing and the Computing Specialisations. It can also be taken as an Advanced Elective subject in the Master of Engineering (Software).</td></tr>
</table>

INDICATIVE CONTENT

Topics are drawn from the field of advanced artificial intelligence including:

- Agent modelling of goals, roles and organisations
- Sensing and acting
- Classical planning
- Theorem proving
- Introduction to modal logic and model theory
- Epistemic logic
- Dynamic logic
- The situation calculus
- Logical regression
- Non-deterministic programming
- Concurrency and multi-agent programming.

Learning Outcomes:

INTENDED LEARNING OUTCOMES (ILO)

On completion of this subject the student is expected to:

1. Understand the theoretical concepts of automated planning and reasoning techniques
2. Be able to apply agent modelling techniques to analyse, design and implement a small agent-based system
3. Be able to evaluate, design, and implement automated planning and reasoning techniques
4. Understand the strengths and weaknesses of different automated planning and reasoning approaches for software agents
5. Be able to apply automated planning and concurrent programming techniques to non-trivial distributed problems

Assessment:

One team-based written assignment (10%) of approximately 1000 words between Weeks 5 to 6, involving 2 to 3 team members with each team member contributing approximately 13 - 15 hours or work, including preparation. ILOs 3 to 4 are addressed in these laboratory classes One team-based presentation (10%) of approximately 10 - 15 minutes duration due in Weeks 7 to 9, involving 2 to 3 team members, each member contributing approximately 13 - 15 hours of work. ILO 5 is addressed in this team-based assignment One programming-based assignment (30%) due in Weeks 11 to 12, requiring approximately 35 - 40 hours of work. ILOs 3 to 5 are addressed in this assignment One written 2-hour closed book end-of-semester examination (50%). ILOs 1 to 3 are addressed in the exam. Hurdle requirement: The examination is a hurdle and must be passed to pass the subject.

Prescribed Texts:

Leon Sterling and Kuldar Taveter, The Art of Agent-Oriented Modelling, MIT Press, 2009.

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 of the subject the students should have the following skills:

- Ability to undertake problem identification, formulation, and solution
- Ability to utilise a systems approach to complex problems and to design and operational performance
- Ability to manage information and documentation
- Capacity for creativity and innovation Ability to communicate effectively with both the engineering team and the community at large.

Notes:

LEARNING AND TEACHING METHODS

The subject involves two 1-hour lectures per week followed by a 1 one hour workshop held in a computer laboratory. Weekly readings are assigned from textbooks, and weekly laboratory exercises are assigned. A significant amount of project work is assigned.

	INDICATIVE KEY LEARNING RESOURCES At the beginning of the year, the coordinator will propose textbook(s) on computer graphics and interaction and will be made available through the University Book Shop and library. Students will have access to lecture notes and lecture slides. The subject LMS site also contains links to recommended literature and current survey papers of software agent principles. **CAREERS / INDUSTRY LINKS** The IT industry is a large and steadily growing industry and advanced artificial intelligence techniques such as software agents are increasingly an integral part of the many facets of this industry. The University of Melbourne and Microsoft have created a new teaching innovative, Apps@Melbourne, for the many talented students keen on developing Apps for tablet computers. Students enrolled in this subject have the opportunity to publish Apps they have developed on the store to be made available to the wider community.
Related Course(s):	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
Related Majors/Minors/ Specialisations:	B-ENG Software Engineering stream MIT Computing Specialisation MIT Distributed Computing Specialisation Master of Engineering (Software)