

# COMP20003 Algorithms and Data Structures

<b>Credit Points:</b>	12.5									
<b>Level:</b>	2 (Undergraduate)									
<b>Dates &amp; Locations:</b>	2016, Parkville This subject commences in the following study period/s: Semester 2, Parkville - Taught on campus.									
<b>Time Commitment:</b>	Contact Hours: 48 hours, comprising of two 1-hour lectures and one 2-hour workshop per week Total Time Commitment: 170 hours									
<b>Prerequisites:</b>	25 points of university-level mathematics <b>AND</b> one of the following: COMP20006 Programming the Machine <table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>COMP20005 Engineering Computation</td> <td>Semester 1, Semester 2</td> <td>12.50</td> </tr> <tr> <td>COMP10002 Foundations of Algorithms</td> <td>Semester 1, Semester 2</td> <td>12.50</td> </tr> </tbody> </table> <p>VCE Algorithmics units 3/4</p> <p>*Students entering COMP20003 through VCE Algorithmics can take mathematics requirements concurrently.</p>	Subject	Study Period Commencement:	Credit Points:	COMP20005 Engineering Computation	Semester 1, Semester 2	12.50	COMP10002 Foundations of Algorithms	Semester 1, Semester 2	12.50
Subject	Study Period Commencement:	Credit Points:								
COMP20005 Engineering Computation	Semester 1, Semester 2	12.50								
COMP10002 Foundations of Algorithms	Semester 1, Semester 2	12.50								
<b>Corequisites:</b>	None									
<b>Recommended Background Knowledge:</b>	None									
<b>Non Allowed Subjects:</b>	Students cannot enrol in and gain credit for this subject and: <table border="1"> <thead> <tr> <th>Subject</th> <th>Study Period Commencement:</th> <th>Credit Points:</th> </tr> </thead> <tbody> <tr> <td>COMP90038 Algorithms and Complexity</td> <td>Semester 1, Semester 2</td> <td>12.50</td> </tr> <tr> <td>COMP20007 Design of Algorithms</td> <td>Semester 1</td> <td>12.50</td> </tr> </tbody> </table>	Subject	Study Period Commencement:	Credit Points:	COMP90038 Algorithms and Complexity	Semester 1, Semester 2	12.50	COMP20007 Design of Algorithms	Semester 1	12.50
Subject	Study Period Commencement:	Credit Points:								
COMP90038 Algorithms and Complexity	Semester 1, Semester 2	12.50								
COMP20007 Design of Algorithms	Semester 1	12.50								
<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>Coordinator:</b>	Dr Nir Lipovetzky									
<b>Contact:</b>	Dr Nir Lipovetzky email: <a href="mailto:nir.lipovetzky@unimelb.edu.au">nir.lipovetzky@unimelb.edu.au</a> (nir.lipovetzky)									
<b>Subject Overview:</b>	<p><b>AIMS</b></p> <p>Programmers can choose between several representations of data. These will have different strengths and weaknesses, and each will require its own set of algorithms. Students will be</p>									

	<p>introduced to the most frequently used data structures and their associated algorithms. The emphasis will be on justification of algorithm correctness, on analysis of algorithm performance, and on choosing the right data structure for the problem at hand. Leading up to an exam with a programming component, quality implementation of algorithms and data structures is emphasized.</p> <p>This subject, or its cognate COMP20007 Design of Algorithms, is a prerequisite for many 300-level subjects in the Computing and Software Systems major.</p> <p><b>INDICATIVE CONTENT</b></p> <p>Topics include: justification of algorithm correctness; asymptotic and empirical analysis of algorithm performance; algorithms for sorting and searching, including fundamental data structures such as trees and hash tables; and graph algorithms.</p>
<b>Learning Outcomes:</b>	<p><b>INTENDED LEARNING OUTCOMES (ILO)</b></p> <p>On completion of this subject the student is expected to:</p> <ol style="list-style-type: none"> <li>1 Present arguments for the correctness or incorrectness of a given algorithm</li> <li>2 Reason about and evaluate the efficiency behaviour of a given algorithm</li> <li>3 Choose appropriate data structures and algorithms for a given problem</li> <li>4 Implement the chosen data structures and algorithms.</li> </ol>
<b>Assessment:</b>	<p>Programming project work during semester, due around weeks 5 and 11, requiring approximately 35 - 40 hours of work (30%) One 30-minute mid-semester test (10%) A 3-hour end-of-semester written examination (60%). Hurdle requirement: To pass the subject, students must obtain at least: 50% overall; 15/30 in project work; 35/70 in the mid-semester test and end-of-semester examination combined. Intended Learning Outcomes (ILOs) 2, 3 and 4 are addressed in all assessment components. ILO 1 is primarily addressed in the second programming project and the final examination.</p>
<b>Prescribed Texts:</b>	TBA
<b>Breadth Options:</b>	<p>This subject potentially can be taken as a breadth subject component for the following courses:</p> <ul style="list-style-type: none"> <li># <b>Bachelor of Arts</b> (<a href="https://handbook.unimelb.edu.au/view/2016/B-ARTS">https://handbook.unimelb.edu.au/view/2016/B-ARTS</a>)</li> <li># <b>Bachelor of Commerce</b> (<a href="https://handbook.unimelb.edu.au/view/2016/B-COM">https://handbook.unimelb.edu.au/view/2016/B-COM</a>)</li> <li># <b>Bachelor of Environments</b> (<a href="https://handbook.unimelb.edu.au/view/2016/B-ENVS">https://handbook.unimelb.edu.au/view/2016/B-ENVS</a>)</li> <li># <b>Bachelor of Music</b> (<a href="https://handbook.unimelb.edu.au/view/2016/B-MUS">https://handbook.unimelb.edu.au/view/2016/B-MUS</a>)</li> </ul> <p>You should visit <b>learn more about breadth subjects</b> (<a href="http://breadth.unimelb.edu.au/breadth/info/index.html">http://breadth.unimelb.edu.au/breadth/info/index.html</a>) and read the breadth requirements for your degree, and should discuss your choice with your student adviser, before deciding on your subjects.</p>
<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 this subject, students should have developed the following skills:</p> <ul style="list-style-type: none"> <li># Application of knowledge of basic science and engineering fundamentals</li> <li># Effective communication about computational efficiency</li> <li># Capacity to reason and solve problems</li> <li># Ability to undertake problem identification, formulation and solution</li> <li># Programming competency</li> <li># Capacity for creativity and innovation</li> <li># Profound respect for truth and intellectual integrity, and for the ethics of scholarship.</li> </ul>
<b>Notes:</b>	<p><b>LEARNING AND TEACHING METHODS</b></p> <p>This subject involves two 1-hour lectures and one 2-hour workshop each week. Although lectures are largely delivered directly based on the slides, often there are in-class discussions</p>

	<p>and development of computer code. The workshops are a blend of tutorial theory questions and implementation of algorithms and data structures in C.</p> <p><b>INDICATIVE KEY LEARNING RESOURCES</b></p> <p>Students are provided with lecture slides, and sample and scaffolding computer code in the C language.</p> <p><b>CAREERS / INDUSTRY LINKS</b></p> <p>With Big Data at the forefront of modern computing solutions, industry is ever-more focused on efficient computational analysis methods. Software engineers, developers and data analysts will find not only the analysis techniques, but also the fundamental algorithmic design concepts, highly applicable to the handling of significant datasets.</p>
<b>Related Majors/Minors/ Specialisations:</b>	B-ENG Software Engineering stream Computer Science Computer Science Science-credited subjects - new generation B-SCI and B-ENG. Selective subjects for B-BMED