

**DEREE COLLEGE SYLLABUS FOR:**

**ITC 3287 Advanced Object Oriented and Functional Programming**  
(Previously ITC 3387 Data Structures and Analysis of Algorithms)  
(Updated Fall 2021)

**3/1/3**  
**UK LEVEL 5**  
**UK CREDITS: 15**

<b>PREREQUISITES:</b>	ITC 2088 Introduction to Programming ITC 2197 Object Oriented Programming Techniques												
<b>CATALOG DESCRIPTION:</b>	Advanced object-oriented and functional programming concepts; algorithmic and problem-solving techniques; advanced GUI components; event handling; data structures; data persistence; efficiency issues.												
<b>RATIONALE:</b>	The module exposes students to modern software development methodologies and programming principles. Students have the opportunity to design, develop, test and deploy rich client applications, enhancing their understanding of graphical user interface design and their judgement of the effectiveness of different development techniques. An understanding of the basic skills needed in algorithmic design and the interaction between algorithm and data structures in creating efficient code is emphasised.												
<b>LEARNING OUTCOMES:</b>	As a result of taking this course, the student should be able to: <ol style="list-style-type: none"><li>1. Demonstrate understanding of the properties of data structures and select the most appropriate one to solve a computing problem.</li><li>2. Explain on a simple problem how functional programming differs from imperative and object-oriented programming.</li><li>3. Model software requirements using UML.</li><li>4. Design, implement, and test, distributable and maintainable object-oriented GUI applications.</li></ol>												
<b>METHOD OF TEACHING AND LEARNING:</b>	In congruence with the teaching and learning strategy of the college, the following tools are used: <ul style="list-style-type: none"><li>• Lectures and class discussions.</li><li>• Laboratory sessions, involving training and practice in program design and development.</li><li>• Homework programming assignments.</li><li>• Office hours held by the instructor to provide further assistance to students.</li><li>• Use of the Blackboard site to further support communication, by posting lecture notes, assignment instruction, timely announcements, formative quizzes and online submission of assignments.</li></ul>												
<b>ASSESSMENT:</b>	<table border="1"><tr><td colspan="2"><b>Summative:</b></td></tr><tr><td>1<sup>st</sup> assessment: Midterm Examination programming problems</td><td><b>30%</b></td></tr><tr><td>2<sup>nd</sup> assessment: Portfolio of student work and oral assessment</td><td><b>10%</b></td></tr><tr><td>Final assessment: Programming Project, report</td><td><b>60%</b></td></tr><tr><td colspan="2"><b>Formative:</b></td></tr><tr><td>Continuation/finalization of problems tackled during the in-class hands-on sessions</td><td><b>0</b></td></tr></table>	<b>Summative:</b>		1 <sup>st</sup> assessment: Midterm Examination programming problems	<b>30%</b>	2 <sup>nd</sup> assessment: Portfolio of student work and oral assessment	<b>10%</b>	Final assessment: Programming Project, report	<b>60%</b>	<b>Formative:</b>		Continuation/finalization of problems tackled during the in-class hands-on sessions	<b>0</b>
<b>Summative:</b>													
1 <sup>st</sup> assessment: Midterm Examination programming problems	<b>30%</b>												
2 <sup>nd</sup> assessment: Portfolio of student work and oral assessment	<b>10%</b>												
Final assessment: Programming Project, report	<b>60%</b>												
<b>Formative:</b>													
Continuation/finalization of problems tackled during the in-class hands-on sessions	<b>0</b>												

	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 80%; padding: 2px;">Midterm formative</td> <td style="width: 20%; text-align: center; padding: 2px;"><b>0</b></td> </tr> </table> <p>The formative assessments aim to prepare students for the summative assessments and expose them to teamwork.  The 1<sup>st</sup> summative assessment tests the LOs 1, 2.  The 2<sup>nd</sup> summative assessment tests the LOs 1 - 4.  The final summative assessment tests the LOs 1 - 4.</p> <p><i>The final grade for this module will be determined by averaging all summative assessment grades, based on predetermined weights for each assessment. If students pass the <b>final summative assessment</b>, which tests all Learning Outcomes for this module, and the average grade for the module is 40 or above, students are not required to resit any failed assessments.</i></p>	Midterm formative	<b>0</b>
Midterm formative	<b>0</b>		
<b>INDICATIVE READING:</b>	<p><b>REQUIRED READING:</b></p> <ol style="list-style-type: none"> <li>1. Charatan Q., Kans A. (2019), <i>Java in Two Semesters</i>, Springer, e-book ISBN: 978-3-319-99420-8.</li> <li>2. Kendal S., (2019), <i>Object Oriented Programming using C#</i>, Kindle edition (free eBook for students through bookboon)  <a href="https://bookboon.com/en/object-oriented-programming-using-c-sharp-ebook">https://bookboon.com/en/object-oriented-programming-using-c-sharp-ebook</a></li> <li>3. Instructor's notes.</li> </ol> <p><b>RECOMMENDED READING:</b></p> <ol style="list-style-type: none"> <li>1. Deitel P., Deitel H. (2018). <i>Visual C# How to Program</i>, Global Edition, Pearson Education Limited</li> <li>2. Chin S. et al. (2019). <i>The Definitive Guide to Modern Java Clients with JavaFX</i>, Apress, Kindle edition available</li> <li>3. Sturm O. (2011). <i>Functional Programming in C#: Classic Programming Techniques for Modern Projects</i>, 1<sup>st</sup> ed., Wrox</li> <li>4. Richards M., Ford N. (2020). <i>Fundamentals of Software Architecture: An Engineering Approach</i>, 1<sup>st</sup> ed., O'Reilly Media, Kindle edition available.</li> </ol>		
<b>INDICATIVE MATERIAL:</b> (e.g. audiovisual, digital material, etc.)	<p><b>REQUIRED MATERIAL:</b> N/A</p> <p><b>RECOMMENDED MATERIAL:</b> N/A</p>		
<b>COMMUNICATION REQUIREMENTS:</b>	<p>Daily access to the course's site on the College's Blackboard CMS.  Communication using proper written and oral English.  Use of word processor and spreadsheet for documentation of assignments.</p>		
<b>SOFTWARE REQUIREMENTS:</b>	<p>MS-Office  JetBrains-IntelliJ  JetBrains-Clion  Microsoft Visual Studio  Microsoft Visio</p>		
<b>WWW RESOURCES:</b>	<p>Learning C#  <a href="https://www.linkedin.com/learning/learning-c-sharp-3?u=80069458">https://www.linkedin.com/learning/learning-c-sharp-3?u=80069458</a></p> <p>Learning JavaFX GUI Development  <a href="https://www.linkedin.com/learning/learning-javafx-gui-development/provide-a-listview-for-user-selections?u=80069458">https://www.linkedin.com/learning/learning-javafx-gui-development/provide-a-listview-for-user-selections?u=80069458</a></p>		

	OOP with JAVA <a href="https://classroom.udacity.com/courses/ud283">https://classroom.udacity.com/courses/ud283</a>
<b>INDICATIVE CONTENT:</b>	<ol style="list-style-type: none"><li>1. Object-Oriented Programming Principles</li><li>2. Collections and Data Structures<ol style="list-style-type: none"><li>a. Linked Lists</li><li>b. Sets</li><li>c. Maps</li><li>d. Stacks, Queues</li><li>e. Hash Tables</li></ol></li><li>3. Generic and Functional Programming</li><li>4. Modelling user requirements with UML</li><li>5. Event-Driven Programming and Event Handling</li><li>6. Visual Programming and GUI Design</li><li>7. Data Persistence</li><li>8. Testing methodologies and version control</li><li>9. Application packaging and distribution</li></ol>