

<b>DEREE COLLEGE SYLLABUS FOR:</b>									
<b>ITC 2088 INTRODUCTION TO PROGRAMMING</b>									
(Updated Fall 2023)									
<b>3/1.5/3</b>									
<b>UK LEVEL: 4</b>									
<b>UK CREDITS: 15</b>									
<b>PREREQUISITES:</b>	None.								
<b>COREQUISITES:</b>	None.								
<b>CATALOG DESCRIPTION:</b>	Problem solving; problem analysis; top-down design of algorithms; implementation; testing and debugging techniques; documentation. Structured programming language constructs: data types, variables, constants, parameters, input and output, selection, iteration, file handling, arrays, elementary sorting and searching algorithms. Modular programming.								
<b>RATIONALE:</b>	The course introduces students to computer programming concepts with an emphasis on the design and implementation of well-documented, structured and logically correct programs. The course is suitable for students with no or little programming experience, who wish to understand the fundamental principles of programming.								
<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 fundamental programming concepts and algorithmic solutions to basic problems using programming techniques.</li> <li>2. Demonstrate understanding of how to trace source code and correctly predict the results.</li> <li>3. Make use of basic data structures and search/sort algorithms to design, implement, test, and debug programs.</li> <li>4. Develop well documented, structured, and maintainable programs.</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>• Classroom lectures, laboratory practical sessions using various simulations tools and progress meetings.</li> <li>• Office hours held by the instructor to provide further assistance to students.</li> <li>• Use of the Blackboard Learning platform, where instructors post lecture notes, assignment instructions, timely announcements, as well as additional resources.</li> </ul>								
<b>ASSESSMENT:</b>	<p><b>Summative:</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>1<sup>st</sup> assessment: Midterm Exam (hands-on) Programming problems</td> <td style="text-align: center;"><b>30%</b></td> </tr> <tr> <td>2<sup>nd</sup> assessment: Portfolio of student work and oral assessment</td> <td style="text-align: center;"><b>10%</b></td> </tr> <tr> <td>Final assessment: Final exam Short programming problems</td> <td style="text-align: center;"><b>60%</b></td> </tr> </table> <p><b>Formative:</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>In-class and take-home short problems</td> <td style="text-align: center;"><b>0%</b></td> </tr> </table>	1 <sup>st</sup> assessment: Midterm Exam (hands-on) Programming problems	<b>30%</b>	2 <sup>nd</sup> assessment: Portfolio of student work and oral assessment	<b>10%</b>	Final assessment: Final exam Short programming problems	<b>60%</b>	In-class and take-home short problems	<b>0%</b>
1 <sup>st</sup> assessment: Midterm Exam (hands-on) Programming problems	<b>30%</b>								
2 <sup>nd</sup> assessment: Portfolio of student work and oral assessment	<b>10%</b>								
Final assessment: Final exam Short programming problems	<b>60%</b>								
In-class and take-home short problems	<b>0%</b>								

	<p>The formative assessments aim to prepare students for the summative assessments and expose them to teamwork.</p> <p>The 1<sup>st</sup> summative assessment tests the LOs 1, 2, 4.</p> <p>The 2<sup>nd</sup> summative assessment tests the LOs 1-4.</p> <p>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>
<b>INDICATIVE READING:</b>	<p><b>REQUIRED READING:</b></p> <ol style="list-style-type: none"> <li>1. Severance, C. (n.d.). Python for informatics: Exploring information. Instructor notes for the C language.</li> </ol> <p><b>RECOMMENDED READING:</b></p> <ol style="list-style-type: none"> <li>1. Cheng, H. H. (2010). C for engineers and scientists: An interpretive approach. New York: McGraw-Hill Higher Education.</li> <li>2. Malik D.S., (2016). <i>Java Programming: Program Design Including Data Structures</i>, Thomson Course Technology.</li> <li>3. Horstmann Cay S., (2016), <i>Big Java: Late Objects</i>, Wiley. ISBN 978-1-118-08788-6.</li> <li>4. Brian Kernighan, Programming in C- A Tutorial.</li> <li>5. Kernighan, B. W., &amp; Ritchie, D. M. (1988). <i>The C programming language</i>. Englewood Cliffs, NJ: Prentice Hall.</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></p> <ol style="list-style-type: none"> <li>1. Ted Jensen, A tutorial on pointers and arrays in C</li> </ol>
<b>COMMUNICATION REQUIREMENTS:</b>	<p>Daily access to the course's site on the College's Blackboard CMS.</p> <p>Communication using proper written and oral English.</p> <p>Use of word processor and spreadsheet for documentation of assignments.</p>
<b>SOFTWARE REQUIREMENTS:</b>	<p>MS-Office</p> <p>C, Python</p> <p>Microsoft Visual C / C++ or any standard C compiler</p> <p>PyCharm</p>
<b>WWW RESOURCES:</b>	<ul style="list-style-type: none"> <li>• <a href="https://opentechschoo.github.io/python-beginners/en/index.html">https://opentechschoo.github.io/python-beginners/en/index.html</a></li> <li>• <a href="https://linux.die.net/man/1/ctopy">https://linux.die.net/man/1/ctopy</a></li> <li>• <a href="https://www.jetbrains.com/pycharm/">https://www.jetbrains.com/pycharm/</a></li> <li>• <a href="http://www.cprogramming.com/">http://www.cprogramming.com/</a></li> </ul>
<b>INDICATIVE CONTENT:</b>	<ol style="list-style-type: none"> <li>1. Overview of computers and programming languages</li> <li>2. Introduction to programming</li> <li>3. Input and Output</li> <li>4. Control structures</li> <li>5. Arrays and applications</li> <li>6. Modular and procedural programming</li> <li>7. OOP Basics</li> </ol>