DEREE COLLEGE SYLLABUS FOR:

ITC 2088 INTRODUCTION TO PROGRAMMING **3/1.5/3**
(Updated Spring 2025) **UK LEVEL: 4**
UK CREDITS: 15

PREREQUISITES:	None.
COREQUISITES:	None.
CATALOG DESCRIPTION:	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. Implementation in python and C programming languages.
RATIONALE:	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.
LEARNING OUTCOMES:	As a result of taking this course, the student should be able to: 1. Demonstrate understanding of fundamental programming concepts and algorithmic solutions to basic problems using programming techniques. 2. Demonstrate understanding of how to trace source code and correctly predict the results. 3. Make use of basic data structures and search/sort algorithms to design, implement, test, and debug programs. 4. Develop well documented, structured, and maintainable programs.
METHOD OF TEACHING AND LEARNING:	In congruence with the teaching and learning strategy of the college, the following tools are used: • Classroom lectures, laboratory practical sessions using various simulations tools and progress meetings, use of generative AI tools to inform course content. • Office hours held by the instructor to provide further assistance to students. • Use of the Blackboard Learning platform, where instructors post lecture notes, assignment instructions, timely announcements, as well as additional resources.
ASSESSMENT:	**Summative**:

1st assessment: Midterm Exam (hands-on) Programming problems	**30%**
2nd assessment: Portfolio of student work and oral assessment	**10%**
Final assessment: Final exam Short programming problems	**60%**

Formative:

In-class and take-home short problems	**0%**

	The formative assessments aim to prepare students for the summative assessments and expose them to teamwork. The 1st summative assessment tests the LOs 1, 2, 4. The 2nd summative assessment tests the LOs 1-4. The final summative assessment tests the LOs 1-4. *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 **final summative assessment,** 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.*
INDICATIVE READING:	**REQUIRED READING:** 1. Severance, C. (n.d.). Python for informatics: Exploring information. Instructor notes for the C language. **RECOMMENDED READING:** 1. Wilson, K. (2022). The Absolute Beginner's Guide to Python Programming. Apress. 2. Cheng, H. H. (2010). C for engineers and scientists: An interpretive approach. New York: McGraw-Hill Higher Education. 3. Brian Kernighan, Programming in C- A Tutorial. *4.* Kernighan, B. W., & Ritchie, D. M. (1988). *The C programming language.* Englewood Cliffs, NJ: Prentice Hall.
INDICATIVE MATERIAL: *(e.g. audiovisual, digital material, etc.)*	**REQUIRED MATERIAL:** N/A **RECOMMENDED MATERIAL:** 1. Ted Jensen, A tutorial on pointers and arrays in C
COMMUNICATION REQUIREMENTS:	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.
SOFTWARE REQUIREMENTS:	MS-Office C, Python Microsoft Visual C / C++ or any standard C compiler (latest IDE) PyCharm (latest IDE)
WWW RESOURCES:	• https://opentechschool.github.io/python-beginners/en/index.html • https://linux.die.net/man/1/ctopy • https://www.jetbrains.com/pycharm/ • http://www.cprogramming.com/
INDICATIVE CONTENT:	1. Overview of computers and programming languages 2. Introduction to programming 3. Input and Output 4. Control structures 5. Arrays and applications 6. Modular and procedural programming 7. OOP Basics