COURSE SPECIFICATION DOCUMENT

Academic School / Department: Science, Innovation & Technology

Programme: MSc Applied Computer Science (Conversion)

***FHEQ* Level:** 7

Course Title: Programming Principles and Concepts

Course Code: COMP 7111

Total Hours: 200 (Lev 7) (4 US Credits)
Timetabled Hours: 39
Guided Learning Hours 21
Independent Learning Hours: 140

Credit 20 UK CATS credits
10 ECTS credits
4 US credits

Course Description:
The course introduces students from non-computing backgrounds to the fundamental principles of programming and computational thinking. It develops practical skills in writing, testing and debugging code, alongside problem-solving abilities. Also, fundamental data structures and basic algorithmic reasoning are also covered thus enabling students to consider efficiency and structural design in software development.

Prerequisites:
None.

Aims and Objectives:

- Introduce foundational programming concepts.
- Develop skills in writing, testing and debugging code.
- Build confidence in computational problem-solving.
- Promote good programming practice and documentation.
- Develop understanding of data abstraction, fundamental data structures and algorithmic reasoning.

Programme Outcomes:
A1, A2, B1, B2, C1, D3

A detailed list of the programme outcomes are found in the Programme Specification. This is located at the archive maintained by Registry and found at: https://www.richmond.ac.uk/programme-and-course-specifications/

Learning Outcomes:
On successful completion of this course students will be able to:

1. Explain and apply core programming concepts including variables, data types and control structures. (A1, A2)
2. Design and implement basic software solutions using modular techniques. (C1, B1)
3. Apply computational thinking to analyse and break down problems. (B1, B2)
4. Write clear and maintainable code following recognised conventions. (C1, D3)
5. Debug and test programs using appropriate tools. (B2, C1)
6. Analyse and evaluate fundamental data structures and basic algorithmic approaches, including consideration of efficiency and abstraction, in the design of software solutions. (A1, A2)

Indicative Content:

- Programming environments and tools
- Software Development Life-Cycle
- Variables, data types and expressions
- Control flow structures
- Functions and modular program design
- Data structures (lists, dictionaries, sets, tuples)
- Input/output handling
- Debugging and testing techniques
- Abstract data types and data abstraction
- Comparative consideration of common data structures and their trade-offs
- Introduction to algorithmic efficiency and basic complexity measures
- Version control fundamentals

Assessment:
This course conforms to the Richmond University Standard Assessment Norms approved at Academic Board and are located at https://www.richmond.ac.uk/university-policies/

Teaching Methodology:
Teaching includes workshops, hands-on programming labs, guided learning, problem-solving sessions and independent practice supported by online materials.

Indicative Text(s):

- Matthes, E. (2023). *Python Crash Course: A Hands-On, Project-Based Introduction to Programming*. 3rd edn. San Francisco: No Starch Press.
- Zelle, J.M. (2024). *Python programming: an introduction to computer science*. Fourth edition. Portland: Franklin.
- Dooley, J. and Kazakova, V.A. (2024). *Software development, design, and coding: with patterns, debugging, unit testing, and refactoring*. Third edition. New York, NY: Apress.

- Weisfeld, M.A. (2019). *The object-oriented thought process*. Fifth edition. Boston, MA: Addison-Wesley.

Websites

- Real Python, *Python Tutorials – Real Python*, available at: https://realpython.com (Accessed: December 2025).
- W3Schools, *Python Tutorial*, available at: https://www.w3schools.com/python (Accessed: December 2025).

See syllabus for complete reading list.

Change Log for this CSD:

Nature of Change	Date Approved & Approval Body (School or AB)	Change Actioned by Registry Services
Guided Learning Hours menu updated	October 2025	
Total Hours Updated	October 2025	