1. **Title of the module**

COMP8840 (CO884) - Algorithms and Logic

1. **School or partner institution which will be responsible for management of the module**

School of Computing

1. **The level of the module (Level 4, Level 5, Level 6 or Level 7)**

Level 7

1. **The number of credits and the ECTS value which the module represents**

15 credits (7.5 ECTS)

1. **Which term(s) the module is to be taught in (or other teaching pattern)**

Spring

1. **Prerequisite and co-requisite modules**

Co-requisite: COMP8810 Object-Oriented Programming

and COMP8820 Advanced Object-Oriented Programming

or COMP8710 Advanced Java for Programmers

1. **The programmes of study to which the module contributes**

Portfolio of Taught Postgraduate Programmes in Computing including:
Computer Science
Computer Science (Artificial Intelligence)
Computer Science (Cyber Security)

1. **The intended subject specific learning outcomes.**
**On successfully completing the module students will be able to:**
	1. Understand a problem description, and then identify, compare, and apply appropriate algorithms to solve it.
	2. Demonstrate ability to formulate formal solutions to problems logically and in algorithmic form using pseudocode.
	3. Reason about the correctness and runtime of algorithms, and reflect on how changes in these algorithms would impact their correctness and runtime.
	4. Evaluate different ways to implement an algorithm, and implement it as part of an executable program.
	5. Understand and implement basic data structures (e.g., arrays, lists, trees) and use algorithmic techniques (recursion and divide & conquer) to solve well-known problems (searching and sorting) and to newly encountered problems.
	6. Understand and manipulate a variety of logical formalisms (e.g., propositional and predicate logic).
	7. Formulate statements and problems in logical form (e.g., as SAT instances).
	8. Understand, at least at a high level, one or more established techniques for automated reasoning and the algorithms involved (e.g., the Davis-Putman-Logemann-Loveland algorithm).
2. **The intended generic learning outcomes.**
**On successfully completing the module students will be able to:**
	1. Communicate information, ideas, problem, and solutions to both specialist and non-specialist audiences.
	2. Critically evaluate arguments, assumptions, abstract concepts and data (that may be incomplete), to make judgements, and to frame appropriate questions to achieve a solution to a problem.
	3. Apply the methods and techniques that they have learned to review, consolidate, extend and apply their knowledge and understanding, and to initiate and carry out projects.
3. **A synopsis of the curriculum**

This module aims to strengthen the foundational programming-in-the-small abilities of students via a strong, practical problem-solving focus. Specific topics will include introductory algorithms, algorithm correctness, algorithm runtime, as well as big-O notation. Essential data structures and algorithmic programming skills will be covered, such as arrays, lists and trees, searching and sorting, recursion, and divide and conquer. One part of the module will also introduce students to logical formalisms such as propositional and predicate logic and some of their applications in program development.

1. **Reading list (Indicative list, current at time of publication. Reading lists will be published annually)**

Steven Skiena, “The Algorithm Design Manual”, Springer, 2008.

Huth, M. & Ryan M., *Logic in computer science: modelling and reasoning about systems*, Cambridge University Press, 2004.

1. **Learning and teaching methods**

Total contact hours: 32

Private study hours: 118

Total study hours: 150

1. **Assessment methods**

13.1 Main assessment methods

Written exercises (25%)

Take-home programming tests (25%)

Examination, 2 hours (50%)

13.2 Reassessment methods

Like for like.

1. **Map of module learning outcomes (sections 8 & 9) to learning and teaching methods (section12) and methods of assessment (section 13)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Module learning outcome** | 8.1 | 8.2 | 8.3 | 8.4 | 8.5 | 8.6 | 8.7 | 8.8 | 9.1 | 9.2 | 9.3 |  |
| **Learning/ teaching method** |  |  |  |  |  |  |  |  |  |  |  |  |
| Lectures | x | x | x | x | x | x | x | x | x | x | x |  |
| Classes | x |  | x | x | x | x | x | x | x | x | x |  |
| Private study | x | x | x | x | x | x | x | x | x | x | x |  |
| **Assessment method** |  |  |  |  |  |  |  |  |  |  |  |  |
| Written exercises | x | x | x | x | x | x | x | x | x | x | x |  |
| Take-home tests | x |  |  | x | x |  |  |  |  |  | x |  |
| Examination |  | x | x |  | x | x | x | x | x | x |  |  |

1. **Inclusive module design**

The School recognises and has embedded the expectations of current equality legislation, by ensuring that the module is as accessible as possible by design. Additional alternative arrangements for students with Inclusive Learning Plans (ILPs)/declared disabilities will be made on an individual basis, in consultation with the relevant policies and support services.

The inclusive practices in the guidance (see Annex B Appendix A) have been considered in order to support all students in the following areas:

a) Accessible resources and curriculum

b) Learning, teaching and assessment methods

1. **Campus(es) or centre(s) where module will be delivered**

Canterbury

1. **Internationalisation**

The topics addressed by this module relate to a field which is of international importance, given the global role of computers in today's technological innovation. The topics covered by this module are international in nature, being identical worldwide and independent of traditional spoken language.

**If the module is part of a programme in a Partner College or Validated Institution, please complete sections 18 and 19. If the module is not part of a programme in a Partner College or Validated Institution these sections can be deleted.**

1. **Partner College/Validated Institution**
2. **University School responsible for the programme**

**DIVISIONAL USE ONLY**

**Revision record – all revisions must be recorded in the grid and full details of the change retained in the appropriate committee records.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date approved | Major/minor revision | Start date of delivery of revised version | Section revised | Impacts PLOs (Q6&7 cover sheet) |
|  |  |  |  |  |
|  |  |  |  |  |