Module code: MOD007357	Version: 4 Date Amended: 12/Jul/2023

1. Module Title

Algorithm Analysis and Data Structures

2a. Module Leader

Ian van der Linde

2b. School

School of Computing and Information Sciences

2c. Faculty

Faculty of Science and Engineering

3a. Level

5

3b. Module Type

Standard (fine graded)

4a. Credits

15

4b. Study Hours

150

5. Restrictions

Type	Module Code	Module Name	Condition
Pre-requisites:	None		
Co-requisites:	None		
Exclusions:	None		
Courses to which this module is restricted:			

LEARNING, TEACHING AND ASSESSMENT INFORMATION

6a. Module Description

Data Structures and Algorithms is described in the ACM/IEEE Joint Task Force for Computing Curricula as being "Fundamental to computer science and software engineering" which also notes that "Algorithms are essential in all advanced areas of computer science: artificial intelligence, databases, distributed computing, graphics, networking, operating systems, programming, security, and so on". In this module you will examine the core data structures and algorithms used in all nontrivial software, enabling you to make sound decisions in the construction of computing solutions that have specific constraints in terms of time (speed) and space (memory). You will learn how to compare the asymptotic behaviour of fundamental computational structures and algorithms and develop the critical skill of making evidence-based choices when selecting from among multiple possible approaches to a given computational problem. To accomplish this, you will study the core mathematical concepts that provide a framework for computational and analytical thinking independently of any particular programming language or computing architecture. In a highly cited cover article by the IEEE Computer Society "What knowledge is important to be a software professional?" the results of a survey of 186 software professionals are presented in which they were asked which topics in Computer Science degree programmes they believed to be the most important. Data Structures & Algorithms was rated the second most important topic, preceded only knowledge of "specific programming languages". The importance of this module to your future career in software development or technical/scientific computing cannot be overemphasised.

6b. Outline Content

Lists, Stacks, Queues, Vectors, Tables, Trees, Heaps, Graphs, Algorithm Types (Brute Force, Backtracking, Divide and Conquer, Dynamic, Greedy, Randomised, Heuristic, Recursive), Sorting, Searching, Complexity (Time, Space, Classes), Optimisation.

6c. Key Texts/Literature

The reading list to support this module is available at: https://readinglists.aru.ac.uk/

6d. Specialist Learning Resources

Latest Releases of: GNU C/C++ Integrated Development Environment (IDE) and Compiler Collection; GNU Multi Precision Arithmetic Library (GMP); MATLAB Numerical Computing Environment and Programming Language.

7. Learning Outcomes (threshold standards)

No.	Type	On successful completion of this module the student will be expected to be able to:
1	Knowledge and Understanding	Analyse and select appropriate data structures and algorithms to support the implementation of an elegant and efficient software artefact.
2	Knowledge and Understanding	Evaluate the behaviour of algorithms to ensure appropriately stable performance as data increases.
3	Intellectual, practical, affective and transferrable skills	Use the features of a high level language to design and implement suitable data structures and associated operations to construct software artefacts of moderate size.
4	Intellectual, practical, affective and transferrable skills	Apply algorithmic complexity metrics to ensure that computational efficiency, memory consumption, and testability constraints are adhered to.

8a. Module Occurrence to which this MDF Refers

Year	Occurrence	Period	Location	Mode of Delivery
2025/6	ZZF	Template For Face To Face Learning Delivery		Face to Face

8b. Learning Activities for the above Module Occurrence

Learning Activities	Hours	Learning Outcomes	Details of Duration, frequency and other comments
Lectures	12	1-4	Lecture 1 hr x 12 weeks
Other teacher managed learning	24	1-4	Tutorial session 2 hr x 12 weeks
Student managed learning	114	1-4	On-line course materials and supplementary reading supporting the lecture and tutorial series are provided on Canvas.
TOTAL:	150		

9. Assessment for the above Module Occurrence

Assessment No.	Assessment Method	Learning Outcomes	Weighting (%)	Fine Grade or Pass/Fail	Qualifying Mark (%)
010	Practical	1-4	50 (%)	Fine Grade	30 (%)

In-class test (1.5 hours)

Assessment No.	Assessment Method	Learning Outcomes	Weighting (%)	Fine Grade or Pass/Fail	Qualifying Mark (%)
011	Practical	1-4	50 (%)	Fine Grade	30 (%)

In-class test (1.5 hours)

In order to pass this module, students are required to achieve an overall mark of 40% (for modules at levels 3, 4, 5 and 6) or 50% (for modules at level 7*).

In addition, students are required to:
(a) achieve the qualifying mark for each element of fine graded assessment as specified above
(b) pass any pass/fail elements

[* the pass mark of 50% applies for all module occurrences from the academic year 2024/25 – see Section 3a of this MDF to check the level of the module and Section 8a of this MDF to check the academic year]