Module Descriptor

Title	Data Structures & Algorithms		
Session	2025/26	**Status**	Published
Code	COMP08106	**SCQF Level**	8
Credit Points	20	**ECTS (European Credit Transfer Scheme)**	10
School	**Computing, Engineering and Physical Sciences**		
Module Co-ordinator	Rebecca Redden		

Summary of Module

Choosing appropriate data structures and algorithms when designing software solutions aids in optimizing performance, reducing time complexity, and enhancing the scalability and maintainability of the system. It allows for efficient storage and retrieval of data, facilitates faster processing, and enables the implementation of complex operations with minimal resource consumption. By understanding and applying foundational concepts in data structures and algorithms, students gain critical problem-solving skills essential for developing robust and efficient software solutions across various domains and applications.

This module provides a comprehensive overview of fundamental data structures such as arrays, linked lists, stacks, queues, trees, and graphs, along with essential algorithms for searching, sorting, and traversing these structures.

Using interactive lectures, practical implementations in various programming languages and hands-on exercises, students will develop a deep understanding of how to analyse problems, select appropriate data structures and algorithms, and design efficient solutions to address real-world challenges in software development.

The syllabus includes:

- Introduction to Data Structures and Algorithms:

o Abstract Data Types, Collection classes in different programming languages

o Relationship to OOP

o Representing Algorithms: Pseudocode, Flow Charts

o Efficiency measures for time and space: rates of growth; asymptotic behaviour, big-O notation.

- Data Structures:

o Basic Structures: Arrays (1 and n-dimensional, jagged) lists (single, double and circularly linked), stacks, queues

o Trees: binary, splay and tree traversing

o Graphs: BFS, DFS, shortest paths

o Dictionary-based data structures: hash tables, maps, sets

- Algorithms:

o Sorting: Linear, Merge, Bubble, Heap, Quick

- o Searching: AVL and Red-Black Trees
- Recursion:
- o Definition
- o Call stack
- o Dynamic programming, divide and conquer

This module will work to develop a number of the key 'I am UWS' Graduate Attributes to make those who complete this module:

- Universal: analytical; ethically-minded; and research-minded
- Work Ready: problem-solver; knowledgeable; and an effective communicator
- Successful: innovative; transformational; and creative

Module Delivery Method	**On-Campus**[1] ☒	**Hybrid**[2] ☐	**Online**[3] ☒	**Work -Based Learning**[4] ☐
Campuses for Module Delivery	☐ Ayr ☐ Dumfries	☒ Lanarkshire ☐ London ☐ Paisley	☒ Online / Distance Learning ☐ Other (specify) Online Delivery / Distance Learning applies to delivery in the BSc (Hons) Data, AI and Software Engineering programme only	

Terms for Module Delivery	Term 1	☐	Term 2	☒	Term 3	☐
Long-thin Delivery over more than one Term	Term 1 – Term 2	☐	Term 2 – Term 3	☐	Term 3 – Term 1	☐

Learning Outcomes	
L1	Demonstrate knowledge of the fundamental data structures and algorithms commonly used in computing science
L2	Assess and articulate the strengths and weaknesses of various data structures across different problem settings

[1] Where contact hours are synchronous/ live and take place fully on campus. Campus-based learning is focused on providing an interactive learning experience supported by a range of digitally-enabled asynchronous learning opportunities including learning materials, resources, and opportunities provided via the virtual learning environment. On-campus contact hours will be clearly articulated to students.

[2] The module includes a combination of synchronous/ live on-campus and online learning events. These will be supported by a range of digitally-enabled asynchronous learning opportunities including learning materials, resources, and opportunities provided via the virtual learning environment. On-campus and online contact hours will be clearly articulated to students.

[3] Where all learning is solely delivered by web-based or internet-based technologies and the participants can engage in all learning activities through these means. All required contact hours will be clearly articulated to students.

[4] Learning activities where the main location for the learning experience is in the workplace. All required contact hours, whether online or on campus, will be clearly articulated to students

L3	Understand the fundamental data structures and algorithms commonly used when developing software systems.
L4	Identify and implement suitable algorithms and data structures for a variety of technically demanding tasks to optimize performance, memory utilization, and scalability while addressing the requirements and constraints of the problem
L5	Evaluate the efficiency and suitability of algorithms in solving specific problems, considering their space and time requirements.

Employability Skills and Personal Development Planning (PDP) Skills	
SCQF Headings	**During completion of this module, there will be an opportunity to achieve core skills in:**
Knowledge and Understanding (K and U)	**SCQF 8** Understanding of algorithm design and data structures and how they fit into the object-oriented programming domain.
Practice: Applied Knowledge and Understanding	**SCQF 8** In using a range of techniques to implement data structures and algorithms for specific problems
Generic Cognitive skills	**SCQF 8** Develop proficiency in implementing data structures and algorithms through active engagement and deep learning Undertake critical analysis of data structure and algorithm concepts and their suitability for a set of routine problems and issues
Communication, ICT and Numeracy Skills	**SCQF 8** Communicate knowledge effectively. Interpret problems and state solutions. Make effective use of tools and information.
Autonomy, Accountability and Working with Others	**SCQF 8** Work on own to gain concepts, identifying their own learning needs.

Prerequisites	**Module Code** COMP07027	**Module Title** Introduction to Programming
	Other or equivalent	
Co-requisites	**Module Code**	**Module Title**

Learning and Teaching
In line with current learning and teaching principles, a 20-credit module includes 200 learning hours, normally including a minimum of 36 contact hours and maximum of 48 contact hours.

Learning Activities During completion of this module, the learning activities undertaken to achieve the module learning outcomes are stated below:	**Student Learning Hours** (Note: Learning hours include both contact hours and hours spent on other learning activities)
Lecture / Core Content Delivery	18
Tutorial / Synchronous Support Activity	6
Laboratory / Practical Demonstration / Workshop	24
Independent Study	152
Please select	
Please select	
TOTAL	200

Indicative Resources
The following materials form essential underpinning for the module content and ultimately for the learning outcomes: The following materials form essential underpinning for the module content and ultimately for the learning outcomes: Bhargava, A.Y. (2024) Grokking Algorithms, Second Edition. Manning. Goodrich, M.T., Tamassia, R. and Goldwasser, M.H. (2018) Data structures and algorithms in Python. New Delhi: Wiley. Sedgewick, R. and Wayne, K. (2011) Algorithms. Upper Saddle River, Nj: Addison-Wesley.
(N.B. Although reading lists should include current publications, students are advised (particularly for material marked with an asterisk*) to wait until the start of session for confirmation of the most up-to-date material)

Attendance and Engagement Requirements
In line with the Student Attendance and Engagement Procedure, Students are academically engaged if they are regularly attending and participating in timetabled on-campus and online teaching sessions, asynchronous online learning activities, course-related learning resources, and complete assessments and submit these on time. **For the purposes of this module, academic engagement equates to the following:** The School of Computing, Engineering and Physical Sciences considers attendance and engagement to mean a commitment to attending, and engaging in, timetabled sessions. You will scan your attendance via the scanners each time you are on-campus and you will login to the VLE several times per week. Where you are unable to attend a timetabled learning session due to illness or other circumstance, you should notify the Programme Leader that you cannot attend. Across the School an 80% attendance threshold is set. If you fall below this, you will be referred to the Student Success Team to see how we can best support your studies.

Equality and Diversity

The University's Equality, Diversity and Human Rights Procedure can be accessed at the following link: UWS Equality, Diversity and Human Rights Code. Aligned with the University's commitment to equality and diversity, this module supports equality of opportunity for students from all backgrounds and learning needs. Using the VLE, material will be presented electronically in formats that allow flexible access and manipulation of content. This module complies with University regulations and guidance on inclusive learning and teaching practice. This module has lab-based teaching and as such you are advised to speak to the Module Co-ordinator to ensure that specialist assistive equipment, support provision and adjustment to assessment practice can be put in place, in accordance with the University's policies and regulations.
(N.B. Every effort will be made by the University to accommodate any equality and diversity issues brought to the attention of the School)

Supplemental Information

Divisional Programme Board	**Computing**
Overall Assessment Results	☐ **Pass / Fail** ☒ **Graded**
Module Eligible for Compensation	☐ **Yes** ☒ **No** **If this module is eligible for compensation, there may be cases where compensation is not permitted due to programme accreditation requirements. Please check the associated programme specification for details.**
School Assessment Board	Business & Applied Computing
Moderator	Bikrant Koirala
External Examiner	A Jindal
Accreditation Details	
Module Appears in CPD catalogue	☐ **Yes** ☒ **No**
Changes / Version Number	1.1

Assessment (also refer to Assessment Outcomes Grids below)
Assessment 1
A formative class test worth 30% of the final mark assessed half way through the module
Assessment 2
A number of small programming tasks given to be submitted evenly spread throughout the module covering various problems within data structures and algorithms introduced (20%).
Assessment 3
A project coursework due at the end of the module (50%)
(N.B. (i) Assessment Outcomes Grids for the module (one for each component) can be found below which clearly demonstrate how the learning outcomes of the module will be assessed. (ii) An indicative schedule listing approximate times within the academic calendar when assessment is likely to feature will be provided within the Student Module Handbook.)

Component 1

Assessment Type	LO1	LO2	LO3	LO4	LO5	Weighting of Assessment Element (%)	Timetabled Contact Hours
Class test (Written)	☒	☒	☒	☐	☒	30	1

Component 2							
Assessment Type	**LO1**	**LO2**	**LO3**	**LO4**	**LO5**	**Weighting of Assessment Element (%)**	**Timetabled Contact Hours**
Portfolio of practical work	☐	☐	☐	☒	☐	20	

Component 3							
Assessment Type	**LO1**	**LO2**	**LO3**	**LO4**	**LO5**	**Weighting of Assessment Element (%)**	**Timetabled Contact Hours**
Creative output/ Audiotapes/ Videotapes/ Games/ Simulations	☐	☒	☒	☒	☒	50	1
Combined total for all components						100%	hours

Change Control

What	When	Who
Attendance and EDI updates	17/01/2025	L Cunningham