

WM180-15 Algorithms and Data Structures

25/26

Department

WMG

Level

Undergraduate Level 1

Module leader

Henry Caushi

Credit value

15

Module duration

30 weeks

Assessment

100% exam

Study location

University of Warwick main campus, Coventry

Description

Introductory description

Algorithms are the fundamental building blocks of computer science – but how can we prove that an algorithm does what we want it to? How can we improve the efficiency of existing algorithms? This module will provide students with a comprehensive understanding of the fundamental principles and techniques in algorithm design and optimisation. Students will explore a wide range of topics, including sorting, searching and pathfinding algorithms, while evaluating their correctness and efficiency.

Module aims

This module enables students to analyse the fundamental principles underlying computer algorithms and the data structures which underlie them. Students will have a solid grasp of core algorithms, and will be able to design new algorithms (and modify existing ones) to solve unseen computational problems.

Outline syllabus

This is an indicative module outline only to give an indication of the sort of topics that may be covered. Actual sessions held may differ.

Basic algorithms:

- Sorting algorithms: Bubble sort, insertion sort, selection sort, quicksort, mergesort and heapsort
- Searching algorithms: Linear search, binary search, hash table lookup

Complexity analysis:

- Big-O, big-Omega and big-Theta notation
- Time complexity analysis
- Space complexity analysis

Strategies for algorithm design:

- Recursive programming
- Divide and conquer
- Dynamic programming and memoisation
- Heuristics
- Any other useful paradigms

Data structures:

- Primitive data structures: Arrays, stacks and queues
- Compound data structures: Hash tables, linked lists, binary search trees, heaps and priority queues
- Advanced data structures: Red-black trees, 2-3-4 trees, binomial trees and binomial heaps

Graph algorithms:

- Graph representations
- Minimum spanning trees, Kruskal's and Prim's algorithms
- Dijkstra's and A* algorithms
- The max-flow min-cut theorem

Learning outcomes

By the end of the module, students should be able to:

- Understand the importance of various algorithms and data structures in cyber security
- Evaluate the correctness and efficiency of algorithms in different programming contexts
- Identify and fix bugs in existing algorithms
- Design new algorithms or modify existing ones to solve unseen problems

Indicative reading list

Cormen, T. H, Leiserson, C. E., Rivest, R. L and Stein, C., 2022. Introduction to Algorithms, 4th Edition. MIT Press.

Interview Cake. <https://www.interviewcake.com>
HackerRank. <https://www.hackerrank.com/>

Subject specific skills

This module improves students' programming skills, giving them an understanding of the underlying algorithms and data structures. Students will also be able to reason about the efficiency of existing algorithms, recognising bugs in existing code as well as opportunities to improve the efficiency of algorithms.

Transferable skills

Logical reasoning, problem solving and effective communication of ideas (both written and verbal)

Study

Study time

Type	Required
Lectures	18 sessions of 1 hour (12%)
Supervised practical classes	18 sessions of 1 hour (12%)
Private study	54 hours (36%)
Assessment	60 hours (40%)
Total	150 hours

Private study description

Implementing the algorithms taught in lectures, understanding the data structures at an in-depth level and practicing programming interview questions

Costs

No further costs have been identified for this module.

Assessment

You do not need to pass all assessment components to pass the module.

Assessment group B

	Weighting	Study time	Eligible for self-certification
Programming interview	30%	15 hours	No
A programming interview which emulates real job interviews held by tech companies. Students will be given a series of programming questions and are tasked with proposing and implementing solutions in a programming language of their choice. Typically, students will also be tasked with finding ways to improve their algorithm's efficiency. The assessor may assist students during the assessment as well as asking questions to challenge their understanding of the algorithms implemented.			
End-of-module exam	70%	45 hours	No
An open-book exam assessing students' understanding of the algorithms and data structures taught in the module.			

~Platforms - WAS

-
- Online examination: No Answerbook required

Feedback on assessment

For the exam, a per-question breakdown of students' marks, together with any marker comments, may be given.

For the programming interview, feedback will be provided during the interview itself. The assessor may provide nudges to students to prevent them from getting lost during the interview, or suggest where their proposed solutions may be incorrect/suboptimal. Students will also be debriefed afterwards, getting an understanding of their strengths and weaknesses.

[Past exam papers for WM180](#)

Availability

Courses

This module is Core for:

- UWMA-H651 Undergraduate Cyber Security
 - Year 1 of H651 Cyber Security
 - Year 1 of H651 Cyber Security
 - Year 1 of H651 Cyber Security