

<b>Module Title:</b>	Data Structures and Algorithms
<b>Language of Instruction:</b>	English
<b>Credits:</b>	5
<b>NFQ Level:</b>	6
<b>Module Delivered In</b>	<a href="#">3 programme(s)</a>
<b>Teaching &amp; Learning Strategies:</b>	As well as traditional lectures students will undertake problem sheets on material presented in class using pseudocode for implementation. Small group tutorials will encourage further problem solving and discussion.
<b>Module Aim:</b>	To introduce the student to Data structures and Algorithms, with an emphasis on application to general problem solving.
<b>Learning Outcomes</b>	
<i>On successful completion of this module the learner should be able to:</i>	
LO1	Design and implement Abstract Data Types including lists, stacks, queues, rings, maps, etc.. and their corresponding algorithms;
LO2	Implement a variety of structures and algorithms for storing, searching and sorting data;
LO3	Outline a range of algorithms for the basic data structures in the areas of graph theory.
<b>Pre-requisite learning</b>	
<b>Module Recommendations</b>	
<i>This is prior learning (or a practical skill) that is recommended before enrolment in this module.</i>	
No recommendations listed	
<b>Incompatible Modules</b>	
<i>These are modules which have learning outcomes that are too similar to the learning outcomes of this module.</i>	
No incompatible modules listed	
<b>Co-requisite Modules</b>	
No Co-requisite modules listed	
<b>Requirements</b>	
<i>This is prior learning (or a practical skill) that is mandatory before enrolment in this module is allowed.</i>	
First year programming or equivalent Specifically Array storage	

**Module Content & Assessment**

**Indicative Content**

**Array Algorithms**

Review of vectors and matrices, basic array algorithms

**Data Structures and Algorithms**

Lists, stacks, queues, rings, maps, Hash tables and collision strategies

**Basic Graph Theory**

Definitions, Representing and applications of simple graphs, Graph theory algorithms.

**Sorting Algorithms**

Simple sorting, bubble, selection, insertion, radix algorithms

**Searching Algorithms**

Linear, binary and hashing. Also Graph searching.

**Assessment Breakdown**

	%
Continuous Assessment	20.00%
Project	80.00%

**Continuous Assessment**

Assessment Type	Assessment Description	Outcome addressed	% of total	Assessment Date
Other	Class test either online or in-class	1,2	20.00	Week 4

**Project**

Assessment Type	Assessment Description	Outcome addressed	% of total	Assessment Date
Project	Group project to design, implement and code the structures and algorithms of a given application (e.g. A Car wash application) and presented in report format	1,2,3	50.00	Week 9
Project	Take Home sheet : to solve a set of unseen problems. Will involve choosing and applying appropriate data structures and algorithms.	1,2,3	30.00	Week 12

No Practical

No End of Module Formal Examination

SETU Carlow Campus reserves the right to alter the nature and timings of assessment

**Module Workload**

<b>Workload: Full Time</b>		
<i>Workload Type</i>	<i>Frequency</i>	<i>Average Weekly Learner Workload</i>
Lecture	12 Weeks per Stage	2.00
Tutorial	12 Weeks per Stage	1.00
Estimated Learner Hours	15 Weeks per Stage	5.93
Total Hours		125.00

**Module Delivered In**

Programme Code	Programme	Semester	Delivery
CW_KCSOF_B	<a href="#">Bachelor of Science (Honours) in Software Development</a>	3	Mandatory
CW_KCSOF_D	<a href="#">Bachelor of Science in Software Development</a>	3	Mandatory
CW_KCCOM_C	<a href="#">Higher Certificate in Science in Computing Programming</a>	3	Mandatory