

## Module specification

When printed this becomes an uncontrolled document. Please access the **Module Directory** for the most up to date version by clicking on the following link: [Module directory](#)

Module Code	COM397
Module Title	Maths and Computing for Problem Solving
Level	3
Credit value	20
Faculty	FACE
HECoS Code	101029
Cost Code	GACP

### Programmes in which module to be offered

Programme title	Is the module core or option for this programme
STEM Foundation Year	Optional

### Pre-requisites

None

### Breakdown of module hours

Learning and teaching hours	24 hrs
Placement tutor support	0 hrs
Supervised learning e.g. practical classes, workshops	16 hrs
Project supervision (level 6 projects and dissertation modules only)	0 hrs
<b>Total active learning and teaching hours</b>	<b>40 hrs</b>
Placement / work based learning	0 hrs
Guided independent study	160 hrs
<b>Module duration (total hours)</b>	<b>200 hrs</b>

<b>For office use only</b>	
Initial approval date	4 Sept 2024
With effect from date	Sept 2024
Date and details of revision	
Version number	1

## Module aims

To integrate mathematical principles and programming skills in an engaging and practical manner, enhancing students' ability to solve real-world problems in computing, engineering, and the built environment. This module aims to build confidence in using mathematical concepts and computational thinking, fostering a problem-solving mindset.

## Module Learning Outcomes - at the end of this module, students will be able to:

1	Understand units, fractions, indices, logarithms and polynomials; also, use algebra to manipulate equations and to solve geometric problems including distance, midpoint, slope, perimeters, areas and volumes.
2	Use trigonometric principles, including Pythagoras' theorem, sine, cosine, and tangent ratios, apply the sine and cosine rules and understand the ratio pi.
3	Represent data in various formats (tables, graphs, charts) and analyse the data using standard statistical methods, including mean, median, mode, standard deviation, etc..
4	Design, write, and debug programs that solve mathematical problems and simulate real-world scenarios, demonstrate an understanding of algorithms and computational thinking.

## Assessment

Indicative Assessment Tasks:

The assessment tasks for this module evaluate students' comprehension and application of mathematical concepts and programming skills through a portfolio and an in-class test. Students solve problems using trigonometry, equation manipulation, programming, and basic statistics. The portfolio requires solving problems using traditional methods and programming, with a reflective commentary. The in-class test assesses overall understanding across the module topics, aiming to develop mathematical and programming proficiency for real-world challenges in computing, engineering, and the built environment.

Assessment number	Learning Outcomes to be met	Type of assessment	Weighting (%)
1	1, 2	In-class test	40
2	3, 4	Portfolio	60

## Derogations

None

## Learning and Teaching Strategies

Aligned with the principles of the Active Learning Framework (ALF), the module will incorporate a blended digital approach utilising a Virtual Learning Environment (VLE). These



resources may include a range of content such as first and third-party tutorials, instructional videos, supplementary files, online activities, and other relevant materials to enhance their learning experience.

The module will be delivered through a combination of lectures, tutorials, practical and programming sessions. Interactive workshops will facilitate hands-on learning, while case studies and project work will allow students to apply their knowledge to real-world scenarios. Access to lecture materials and additional resources will be provided via the University's VLE platform.

## Indicative Syllabus Outline

---

Number Systems and Data Representation

Units, Fractions, Indices, Logarithms and Polynomials.

Algebra: Rearranging and Manipulating Equations.

Geometry: Distances, Midpoints, Perimeters, Areas and Volumes. Simpsons and Mid-Ordinant rules.

Trigonometry: Pythagoras' Theorem. Sine, Cosine and Tangent ratios. Sine and Cosine rules. Circles, the ratio Pi, Circumference and Area.

Basic Statistics and Data Analysis – Mean, Median, Mode, Inter-Quartile Ranges and Standard Deviations. Graphical Representation – Bar Charts, Histograms, Scatter Graphs.

Algorithms and Computational Thinking.

Problem-Solving with Programming.

## Indicative Bibliography:

---

Please note the essential reads and other indicative reading are subject to annual review and update. Please *ensure correct referencing format is being followed as per University Harvard Referencing Guidance.*

### Essential Reads

Croft, A. and Davison, R. (2016), *Foundation Maths*, 6<sup>th</sup> Ed. Prentice Hall.

### Other indicative reading

Stroud, K.A, Booth, D.J (2009), *Foundation Mathematics*, 1<sup>st</sup> Ed. Palgrave.

Jenkyns, T. & Stephenson, B. (2018), *Fundamentals of Discrete Math for Computer Science*. 2<sup>nd</sup> Ed. Springer