COURSE DETAILS
Course Code: F78AP
Full Course Title: Algorithmic and Scientific Programming
SCQF Level: 8
SCAF Credits: 7.5
Available as Elective: No

DELIVERY LEVEL
Undergraduate: Yes  Postgraduate Taught: No  Postgraduate Research: No

Additional Information:

COURSE AIMS
• To introduce the use of algorithms to solve computational problems
• To equip students with fundamentals of computer programming; skills and techniques which may be applied in a wide variety of programming languages
• To have students create algorithms to solve computational problems in at least two programming languages used in modern data science (such as R and Python)

LEARNING OUTCOMES – SUBJECT MASTERY
At the end of this course, students should be able to:

• design and implement appropriate algorithms to solve computational problems
• understand the logical operation of computer programmes
• use appropriate control structures, loops and conditional statements within computer programmes
• use appropriate data structures within computer programmes

At the end of this course, students should be able to:

• systematically analyse the structure and operation of unfamiliar algorithms and computer programmes
• design algorithms and programmes to solve unfamiliar computational problems

LEARNING OUTCOMES – PERSONAL ABILITIES
At the end of this course, students should be able to:

• use appropriate algorithms and computer programmes to solve problems relevant to modern data science and related industries

At the end of this course, students should be able to:
F78AP Algorithmic and Scientific Programming

• apply their understanding of computer programming to independent study of unfamiliar programming languages

At the end of this course, students should be able to:

• design and implement appropriate algorithms to solve mathematical and statistical problems
• use appropriate software to write and execute computer programmes
• write computer code understandable to other users

SYLLABUS

• Introduction to computational algorithms
• Conditional statements: if, else
• Loops: for, while
• Writing functions
• Recursion
• Vectors, arrays and matrices
• Graphics

Students will study the above using at least two programming languages (e.g. R and Python)

COURSE RELATIONSHIPS

N/A

LOCATION AND ASSESSMENT METHODS

<table>
<thead>
<tr>
<th>Edi</th>
<th>SBC</th>
<th>Ork</th>
<th>Dub</th>
<th>Malay</th>
<th>IDL</th>
<th>COLL</th>
<th>ALP</th>
<th>OTH</th>
<th>Method</th>
<th>Weight</th>
<th>Exam Mins</th>
<th>Type</th>
<th>Diet</th>
<th>Synoptic Course</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Coursework</td>
<td>100</td>
<td></td>
<td>Assessment</td>
<td>Semester 2</td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Coursework</td>
<td>100</td>
<td></td>
<td>Reassessment</td>
<td>Semester 3</td>
<td></td>
</tr>
</tbody>
</table>