F19NM Scientific Computing

COURSE DETAILS
Course Code: F19NM
Full Course Title: Scientific Computing
SCQF Level: 9
SCAF Credits: 15
Available as Elective: Yes

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

COURSE AIMS
To give an introduction to some of the basic methods of numerical analysis via a widely used scientific computing package.

LEARNING OUTCOMES – SUBJECT MASTERY

- Basic understanding of numerical analysis and the numerical approximation of solutions of mathematical problems.
- Use of mathematical techniques for approximating derivatives, integrals and the solutions of nonlinear equations.
- Ability to approximate and interpolate a function.
- Be able to solve a linear system by standard direct methods.
- Be able to carry out iterative algorithms for the solution of linear systems and eigenvalue problems.
- Appreciate the value of careful analysis of algorithms for efficiency and accuracy.

LEARNING OUTCOMES – PERSONAL ABILITIES

- Ability to use computer software to solve mathematical problems.
- The ability to critically assess sources and types of errors in problems.
- Critical awareness of the power of abstraction in understanding physical situations.
- Ability to use computer simulations to understand abstract systems and approximate real-world problems.
- Organize complex calculations in a clear manner.
- Be aware of the importance of understanding errors.
- Be able to present a written account of technical material.

SYLLABUS

- Introduction to scientific computing software: Basic operations, vectors and matrices, plotting graphs of functions, loops, conditional statements.
- Solution of nonlinear algebraic equations: Approximating solutions of f(x)=0 using, e.g., the bisection, Newton and fixed-point methods.
- Polynomial Interpolation: Approximating functions of one variable by polynomials.
- Numerical Integration: Approximating integrals of functions of one or more variables using, e.g., Newton-Cotes methods, composite quadrature rules, Gaussian quadrature.
• Numerical Differentiation: Approximating derivatives using finite differences, e.g., forward, backward and central difference methods.
• Direct methods for solving linear systems: Approximating solutions of $Ax=b$ using, e.g., Gaussian elimination and LU and Cholesky decompositions.
• Iterative methods for solving linear systems: Approximating solutions of $Ax=b$ using, e.g., Jacobi, Gauss-Seidel, SOR and Krylov subspace methods.
• Iterative methods for solving eigenvalue problems: Approximating eigenvalues and eigenvectors using, e.g., power, inverse power, QR and Krylov subspace methods.

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Level</th>
<th>Title</th>
<th>School</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>F18NA</td>
<td>8</td>
<td>Numerical Analysis A</td>
<td>School of Math and Comp Sci.</td>
<td>Excluded</td>
</tr>
</tbody>
</table>

<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></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Coursework</td>
<td>30</td>
<td>Assessment</td>
<td>Semester 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Examination</td>
<td>70</td>
<td>120</td>
<td>Assessment</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Examination</td>
<td>100</td>
<td>120</td>
<td>Reassessment</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>