COURSE SPECIFICATION FORM

<table>
<tr><td colspan="4">DEPARTMENT OF MATHEMATICS</td><td colspan="2"></td></tr>
<tr><td>Course Code:</td><td>MT5465</td><td>Course Value:</td><td>200 hours</td><td>Status:
(ie:Core, or Optional)</td><td>Optional</td></tr>
<tr><td>Course Title:</td><td colspan="3">Network Algorithms</td><td>Availability</td><td>Term 2</td></tr>
<tr><td>Prerequisites:</td><td colspan="3">An undergraduate course in discrete mathematics</td><td>Recommended:</td><td>None</td></tr>
<tr><td>Aims:</td><td colspan="5">• To introduce the formal idea of an algorithm, when it is a good algorithm and techniques for constructing algorithms and checking that they work.
• To explore connectivity and colourings of graphs, from an algorithmic perspective.
• To show how algebraic methods such as path algebras and cycle spaces may be used to solve network problems.</td></tr>
<tr><td>Learning Outcomes:</td><td colspan="5">On completion of the course the students should be able to:
• use particular algorithms which optimize various properties for graphs and networks and prove that they work;
• understand ideas of complexity exemplified in particular by the Travelling Salesman Problem;
• apply Fleury’s and Tucker’s algorithms to find Eulerian trails;
• find chromatic polynomials and illustrate Vizing’s theorem on edge colourings;
• use path algebra methods to find maximal flows, critical paths and similar problems.</td></tr>
<tr><td>Course Content:</td><td colspan="5">Trees: Algorithms for minimum spanning trees.
Sorting and searching: Sorting methods including bubble sort and heap sort. Depth first search and breadth first search. Shortest paths.
The Travelling Salesman Problem: Branch and bound method, upper and lower bounds, approximate methods.
Flows in networks: The max-flow min-cut theorem. An algorithm for finding maximum flows.
Matching problems: Hall's theorem. Maximum and complete matchings. Alternating paths and applications. Menger's theorems on edge and vertex connectivity.
Eulerian trails: Algorithms for finding them: Fleury's algorithm; Tucker's algorithm.
Hamiltonian paths: Ore’s and Dirac’s theorems on Hamiltonian cycles.
Colouring graphs: Vertex and edge colourings; chromatic polynomials. Brook's, Vizing's and König's theorems. Colouring maps, the four-colour theorem.
Path algebras: Definitions, strong and weak closure, matrices over path algebras, absorptive matrices, applications to various network problems, including critical path analysis
Cycle spaces: Definitions, feasible flows, displacement networks. Maximum flow minimum cost flows, cost-reducing cycles.</td></tr>
<tr><td>Teaching & Learning Methods:</td><td colspan="5">44 hours of lectures and examples classes.
156 hours of private study, including work on problem sheets and examination preparation. This may include discussions with the course leader if the student wishes.</td></tr>
<tr><td>Key Bibliography:</td><td colspan="5">Algorithmic Graph Theory – A. Gibbons (Cambridge UP). Library Ref. 512.23 GIB
Discrete Mathematics – N.L. Biggs (Oxford UP). Library Ref. 510 BIG
Graphs and Networks – B. Carré (Oxford UP). Library Ref. 512.23 CAR</td></tr>
<tr><td>Formative Assessment & Feedback:</td><td colspan="5">Formative assignments in the form of 8 problem sheets. The students will receive feedback as written comments on their attempts.</td></tr>
<tr><td>Summative Assessment:</td><td colspan="5">Exam (%) Four questions out of five in a two-hour paper: 100%
Coursework (%) None
Deadlines: n/a</td></tr>
</table>