COURSE SPECIFICATION FORM

for new course proposals and course amendments

Department/School:	**Mathematics**	**Academic Session:**	**2017-18**
Course Title:	Graphs and Optimisation	**Course Value:** (UG courses = unit value, PG courses = notional learning hours)	0.5 unit
Course Code:	MT2630	**Course JACS Code:** (Please contact Data Management for advice)	G100
Availability: (Please state which teaching terms)		**Status:**	Mandatory for GG41, LG11, NG31, G1N2 Condonable Optional for other programmes
Pre-requisites:	MT1810, MT1820	**Co-requisites:**	-
Co-ordinator:	-		
Course Staff:	-		
Aims:	This course provides an introduction to graph theory and linear programming. Many networks such as railroad networks, social networks, electrical networks and the worldwide-web can be modelled by graphs. In this course we consider graphs and directed graphs and establish some of their basic properties. We then consider flows in networks and see how flows are related to a more general optimisation concept called linear programming. The course then introduces other linear programming problems and how to solve them algorithmically. To every linear programme there exists a dual programme and we investigate the relationship between the solutions of these two problems. Key aims of the course include: to provide an introduction to graphs and linear programming; to develop the skill to formulate and solve linear programmes and their duals.		
Learning Outcomes:	On completion of the course, students should: know the basic notions of graph theory; know basic examples of graph classes, like paths, cycles and trees; be able to formulate problems as matching and flow problems; be able to formulate linear programming problems; be able to solve linear programming problems with the simplex algorithm; be able to formulate the dual of a linear programming problem; know the strong duality theorem.		
Course Content:	Graphs: Definition of a graph and directed graph; basic examples of graphs such as paths, cycles, trees; basic properties of graphs such as graph isomorphism, the handshaking lemma, degree sequences, characterisations of trees, connectivity and components; matchings and flows in graphs; Hall's theorem. Optimisation: Examples of formulation of problems as Linear Programmes; flow problem as an LP problem. Unbounded and infeasible LPs; feasible and basic feasible solutions; the simplex algorithm. LP duality; the weak and strong duality theorem; dual of the dual is the primal; complementary slackness and testing for optimality. The branch and bound method for integer programming.		
Teaching & Learning Methods:	33 hours of lectures and examples classes. 117 hours of private study, including work on problem sheets and examination preparation. This may include discussions with the course leader if the student wishes.		
Key Bibliography:	Modern Graph Theory B.Bollobás Graph Theory R. Diestel Linear Programming, Foundations and Extensions R.J. Vanderbei Linear Programming V. Chvátal		
Formative Assessment & Feedback:	Formative assignments in the form of 8 problem sheets. The students will receive feedback as written comments on their attempts.		
Summative Assessment:	Exam A two hour paper (100%) Coursework (%) None		

Updated September 2017