3/0/3

DEREE COLLEGE SYLLABUS FOR:

CS3413 ALGORITHMS AND COMPLEXITY   - LEVEL 5
 (Updated Spring 2012)

UK CREDITS: 15

PREREQUISITES:	CS1070 Introduction to Information Systems MA1001 Finite Mathematics MA2106 Mathematics for Computing CS2188 Introduction to Programming
CATALOG DESCRIPTION:	Study of algorithms and their complexity. Design, analysis and evaluation of performance. Complexity theory and classes of complexity. O, Big O and Theta notation. Computational models (Turing machine). Union-Find, Divide and Conquer, Greedy strategy, dynamic programming, search in trees and graphs, backtracking, NP-completeness.
RATIONALE:	To course aims to acquaint students with the notion of algorithms and complexity, to teach computational models, to present generalised techniques for the design and analysis of algorithms and to apply techniques for problem solving.
LEARNING OUTCOMES:	As a result of taking this course, the student should be able to: 1. Identify classes of computational complexity. 2. Describe computational models. 3. Analyse algorithms in terms of complexity. 4. Design efficient algorithms for problem solving. 5. Contrast major algorithmic techniques. 6. Solve problems involving graph search.
METHOD OF TEACHING AND LEARNING:	In congruence with the teaching and learning strategy of the college, the following tools are used: • Lectures, class discussions, laboratory practical sessions and problem solving. • Office hours: Students are encouraged to make full use of the office hours of their instructor, where they can ask questions and go over lecture material. • Use of the Blackboard Learning platform, where instructors post lecture notes, assignment instructions, timely announcements, as well as additional resources.

ASSESSMENT:

Summative:

Project: programming problems	**40%**
Final Examination (2-hour comprehensive), essay questions and problem solving	**60%**

Formative:

Coursework : programming problems	**0**

The formative coursework aims to acquaint students with the material and prepare them for the final examination.

The summative project tests learning outcomes 3,4,6
The final examination tests learning outcomes 1-6

INDICATIVE READING:

REQUIRED READING:
Introduction to the Design and Analysis of Algorithms, Anany V. Levitin, 3rd Edition (or latest), Addison Wesley,ISBN-10: 027376411X ISBN-13: 978-0273764113

RECOMMENDED READING:
H. Lewis and C.H. Papadimitriou, "Elements to the Theory of Computations", Prentice Hall

COMMUNICATION REQUIREMENTS:

Daily access to the course's site on the College's Blackboard CMS. Effective presentation skills using proper written and oral English. Communicate and coordinate during team activities.

SOFTWARE REQUIREMENTS:

Java or C programming languages

WWW RESOURCES:

ACM wiki on algorithms and complexity:
http://wiki.acm.org/cs2001/index.php?title=Algorithms_and_complexity

Algorithms wiki: http://en.wikibooks.org/wiki/Algorithms

Complexity theory from Wolfram:
http://mathworld.wolfram.com/ComplexityTheory.html

Information and Computation journal:
http://projects.csail.mit.edu/iandc/

Journal of Graph Algorithms and Applications:
http://www.cs.brown.edu/sites/jgaa/

ACM journal on experimental Algorithms:
http://www.jea.acm.org/about.html

ACM Transactions on Algorithms http://talg.acm.org/

Journal of the ACM http://jacm.acm.org/

The NP versus P source, http://www.win.tue.nl/~gwoegi/P-versus-NP.htm

Algorithms and Complexity resources:

http://www.dcs.gla.ac.uk/research/algorithms/links.html

INDICATIVE CONTENT:

1. The role of algorithms in computing
2. Growth of functions
3. Divide and Conquer
4. Medians and Order Statistics
5. Hash Tables
6. Dynamic Programming
7. Greedy Algorithms
8. Elementary Graph Algorithms
9. Minimum Spanning Trees
10. Shortest Paths
11. Maximum Flow
12. NP-completeness