CSC 612: ANALYSIS OF ALGORITHMS

COURSE CATALOG ENTRY:

An advanced study in computer algorithms that delves deeply into a selected problem: linear programming (simplex, revised simplex, complementary slackness, Khachian's ellipsoid, etc) and duality; in addition, the course also covers suffix trees, minimum spanning trees, Bellman-Ford and Djikstra's shortest paths algorithms, and computational geometry.

TEXTBOOK:

Required Textbook : *Linear Programming*, James P. Ignizio, Tom M. Cavalier

Alternate Textbook : *Introduction to Algorithms, 2nd Ed*, by Cormen, Leiserson, Rivest, Stein

COURSE OBJECTIVES:

This course will take an example problem area and cover it in some depth. Students will be able to handle linear programming problems and understand the concept of duality; analyze graphs (minimum spanning trees, shortest path analysis); and will familiarize students with several computational geometry algorithms.

PREREQUISITES:

CSC-311 (Design and Analysis of Algorithms) (*or equivalent*)

COMMUNICATIONS:

***web*:** http://www.drchip.org/cua/csc612/index.html

***email*:** charles.e.campbell@nasa.gov

Contribution of Course to Meeting the Professional Component:

This course considers an important problem area (linear programming) and develops a number of algorithms which address it. The notion of duality, wherein a difficult problem can be converted to a "dual" problem, solved, and the solution converted back to the original problem, can be applied to physics, electrical problems, networks, etc. Network analysis and computational geometry are used to develop optimal wiring diagrams, optimal flow, etc; the methods of proof given are useful for students' research.

Relationship of Course to Program Objectives:

This course supports several educational objectives of the EE&CS Department as listed:

1. Students are exposed to optimization algorithms (linear programming, duality)
2. Students are introduced to algorithms' proofs
3. Students will be enabled to effectively apply optimization to linear problems, determine shortest paths in networks, determine spanning trees and convex hulls

COURSE OUTLINE:

Although the topic sequence is expected to remain unchanged, our progress may vary.

Time	*Text*	*Topic(s)*
09/02/2016	LP p10-19,42-53	Linear Programming: general form, converting to standard form, graphical solution, extreme points, optimality, basic feasible solutions
09/09/2016	LP p80-111	Simplex Method, entering/departing variables, optimality conditions, checking if unbounded, tableaux, initializing, degenerate solutions, cycling, lexicographic anti-cycling
09/16/2016	LP p80-111	Simplex Method, entering/departing variables, optimality conditions, checking if unbounded, tableaux, initializing, degenerate solutions, cycling, lexicographic anti-cycling
09/23/2016	LP p135-141	Revised Simplex Method
09/30/2016	LP p168-187,197-216	Duality, forming LP dual, relationships, primal/dual, complementary slackness, geometric interpretation, dual variables as rates change, dual simplex algorithm, sensitivity analysis
10/07/2016	LP p168-187,197-216	Duality, forming LP dual, relationships, primal/dual, complementary slackness, geometric interpretation, dual variables as rates change, dual simplex algorithm, sensitivity analysis
10/14/2016	Midterm - 2hrs	
10/21/2016	LP p441-449,420-429	Primal-Dual, Integer Programming
10/28/2016	LP p247-270	Computational complexity, Khachian's ellipsoid, Primal Affine Scaling, Karmarkar's projective
11/04/2016	LP Lab	in class
11/11/2016	Notes	String matching - Brute force, Rabin Karp
11/18/2016	Notes	String matching - Finite State Automata
	Notes	String matching - Aho-Corasick
	Notes	String matching - Knuth-Morris-Pratt
11/25/2016	*no class*	Thanksgiving
11/18/2016	Notes	String matching - Boyer Moore
12/02/2016	ItoA p561-573	Minimum Spanning Trees: Kruskal, Heaps, Prim
12/09/2016	Notes	String matching - Suffix Trees
Time Permitting	ItoA 580-599	Single Source Shortest Paths - Bellman-Ford, Djikstra
	ItoA p933-961	Computational Geometry
	ItoA p620-635	Min Time Routing, All Pairs Shortest Paths
12/16/2016	Final Exam	

Expected Learning Outcomes:

After completing this course, students are expected to:

ELO-1 : Understand linear programming concepts: exteme points, basic and feasible solutions

ELO-2 : Understand and able to perform the simplex method and revised simplex method

ELO-3 : Understand duality, primal/dual relationships, complementary slackness

ELO-4 : Able to perform dual simplex method, primal/dual algorithm, integer programming

ELO-5 : Understand and able to perform suffix tree analysis

ELO-6 : Understand and able to determine minimum spanning trees and determine shortest paths (graph theory)

ELO-7 : Understand and able to determine convex hulls, if a point is inside

Expected Learning Outcome/Program Outcome Matrix:

	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11
ELO-1	X	X	X						X	X	X
ELO-2	X	X	X	X		X	X	X	X	X	X
ELO-3	X	X							X	X	
ELO-4	X	X	X	X		X		X	X	X	
ELO-5	X	X	X	X		X		X	X	X	X
ELO-6	X	X	X						X	X	
ELO-7	X	X								X	X

Outcome Assessment:

This course employs the following mechanisms to assess the above learning outcomes:

- Homework is assigned weekly, due in two weeks; it is used to assess the level of student understanding of topics presented in the lectures.
- In-class one or two question quizzes are given, graded in class, and solutions given, thus providing rapid feedback to students. Student solutions that are 100% correct yield "extra credit" points towards the next exam (midterm, final).
- Student performance on midterm and final exams is used to assess learning outcomes.

Course Improvement:

The professor continually tries to improve the course:

- Student suggestions during the semester are seriously considered (the current slide format is the result of prior student suggestions)
- Extremely important are the self-evaluation surveys students provide which assess their individual learning objective performance.
- At the end of every semester, the teacher meets with the chairman to discuss improvement plans for the course based on the SUMA Student Course Evaluation organized by the University.

Method of Assessment

Homework	20%	given each week, due in two weeks
Lab	10%	lab grades
Midterm	35%	one page of notes, both sides permitted
Final Examination	35%	two pages of notes, both sides permitted