The University of Georgia

Department of Computer Science

Course Information Sheet
CSCI 4670
Combinatorics

Brief Course Description
(50-words or less)

Basic counting principles: permutations, combinations, probability, occupancy problems, and binomial coefficients. More sophisticated methods include generating functions, recurrence relations, inclusion/exclusion principle, and the pigeonhole principle. Additional topics include asymptotic enumeration, Polya counting theory, combinatorial designs, coding theory, and combinatorial optimization.

Extended Course Description / Comments

This course is cross-listed as MATH 4670. This is a 3 credit hour course

Pre-Requisites and/or Co-Requisites

CSCI 2610 Discrete Mathematics for Computer Science or MATH 3200 Intro. To Higher Math

MATH 3000 or MATH 3500 or MATH 3500H

Approved Textbooks
(if more than one listed, the textbook used is up to the instructor's discretion)

Author(s): Richard A. Brualdi
Title: Introductory Combinatorics
Edition: Fifth
ISBN-13: 978-0136020400

Specific Learning Outcomes (Performance Indicators)

This course presents a survey of topics in combinatorics. At the end of the semester, all students will be able to do the following:

1. Apply the sum rule for counting
2. Apply the product rule for counting.
3. Prove identities involving binomial coefficients
4. Derive a recursion from a generating function equation.
5. Compute the probabilities of the five poker hands.
6. State the fundamental theorem of inclusion/exclusion.
7. Demonstrate a Steiner Triple System

Relationship Between Student Outcomes and Learning Outcomes

Learning Outcomes	a	b	c	d	e	f	g	h	i	j	k
1	•	•								•	
2	•	•								•	
3	•	•								•	
4	•	•								•	
5	•	•								•	
6	•	•								•	
7	•	•								•	

Student Outcomes

a. An ability to apply knowledge of computing and mathematics appropriate to the discipline.
b. An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.
c. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.
d. An ability to function effectively on teams to accomplish a common goal.

e. An understanding of professional, ethical, legal, security and social issues and responsibilities.
f. An ability to communicate effectively with a range of audiences.
g. An ability to analyze the local and global impact of computing on individuals, organizations, and society.
h. Recognition of the need for and an ability to engage in continuing professional development.
i. An ability to use current techniques, skills, and tools necessary for computing practice.
j. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
k. An ability to apply design and development principles in the construction of software systems of varying complexity.

Major Topics Covered
(Approximate Course Hours)

3 credit hours = 37.5 contact hours
4 credit hours = 50 contact hours

Note: Exams count as a major topic covered

Propositional logic (3.5-hours)
Predicate logic (3.5-hours)
Proofs: types of proofs (4-hours)
Sets, set logic and set operations (2-hours)
Functions (2-hours)
Sequences and summations (2-hours)
Integer algorithms (3-hours)
Modular arithmetic (.5-hours)
Mathematical induction (3.5-hours)
Counting (2.5-hours)
The pigeonhole principle (.5-hours)
Permutations and combinations (2.5-hours)
Finite probabilities (4-hours)
Relations (2.5-hours)
Using graphs to represent relations (1.5-hours)

Typical problems posed in combinatorial fashion: "how many ways can you ... ?"
2. Pigeonhole principle.
3. Permutations, combinations, Pascal's triangle, multinomial coefficients.
4. Inclusion/exclusion principle, derangements, permutations with forbidden positions.
5. Recurrence relations and generating functions.
6. Special counting numbers: Stirling numbers, Catalan numbers.
7. Applications to block designs, graph theory.

Assessment Plan for this Course

Each time this course is offered, the class is initially informed of the Course Outcomes listed in this document, and they are included in the syllabus. At the end of the semester, an anonymous survey is administered to the class where each student is asked to rate how well the outcome was achieved. The choices provided use a 5-point Likert scale containing the following options: Strongly agree, Agree, Neither agree or disagree, disagree, and strongly disagree. The results of the anonymous survey are tabulated and results returned to the instructor of the course.

The course instructor takes the results of the survey, combined with sample student responses to homework and final exam questions corresponding to course outcomes, and reports these results to the ABET committee. If necessary, the instructor also writes a recommendation to the ABET committee for better achieving the course outcomes the next time the course is offered.

How Data is Used to Assess Program Outcomes	Each course Learning Outcome, listed above, directly supports one or more of the Student Outcomes, as is listed in "Relationships between Learning Outcomes and Student Outcomes". For CSCI 46790, Student Outcomes (a), (b), and (j) are supported.
Course Master	Dr. E. Rodney Canfield
Course History	05/2008 Course Information Uploaded to CAPA 02/2012 Course Information Sheet Created