UNIVERSITY OF WATERLOO

FINAL EXAMINATION

FALL TERM 2002

Surname: ______________________

First Name: ______________________

Id.#: ______________________

Course Number	MATH 239
Course Title	Introduction to Combinatorics
Instructor	01 Goulden 2:30 ☐ 03 Wagner 1:30 ☐ 04 Wagner 10:30 ☐ 05 Schellenberg 9:30 ☐
Date of Exam	December 14, 2002
Time Period	7-10 p.m.
Number of Exam Pages (including this cover sheet)	13 pages
Exam Type	Closed Book

ADDITIONAL INSTRUCTIONS:

1. Write your name and Id.# in the blanks above. Put a check mark in the box next to your instructor's name and lecture time.
2. There are 13 pages to this exam including the cover page. Please be sure you have all 13 pages.
3. Answer each of the problems in the space provided; use the back of the previous page for additional space.

4. You may only use a non-programmable calculator. Show the reasoning used in any calculation.

Problem	Value	Mark Awarded	Problem	Value	Mark Awarded
1	10		6	10	
2	12		7	10	
3	8		8	16	
4	14		9	10	
5	10		TOTAL	100	

1. Let a_n, $n \geq 0$, be the number of compositions of n in which all parts are at least equal to 4, *and* all parts are congruent to 1 (modulo 3).

[7] **(a)** Prove that

$$\sum_{i \geq 0} a_i x^i = \frac{1 - x^3}{1 - x^3 - x^4}.$$

[3] **(b)** Give a linear recurrence equation for the sequence $\{a_n : n \geq 0\}$, and enough initial conditions to determine the sequence uniquely.

[8] **2(a)** Let b_n, $n \geq 0$, be the number of $\{0,1\}$–strings of length n in which every block of 0s has odd length, and every block of 1s has length exactly equal to 1. Prove that

$$\sum_{i\geq 0} b_i x^i = \frac{1+2x-x^3}{1-2x^2}.$$

[4] **(b)** Prove that $2\, b_{2m+1} = 3\, b_{2m}, \;\; m \geq 1.$

[8] **3.** Solve the recurrence equation $c_n = 2\,c_{n-1} + 4\,c_{n-2} - 8\,c_{n-3}$, with initial conditions $c_0 = 1,\ c_1 = 1,\ c_2 = 1.$

4. Let A_n, $n \geq 1$, be the graph whose vertices are the subsets of size n chosen from the set $\{1, 2, \ldots, 2n+2\}$, where two subsets are adjacent if they are disjoint (i.e., their intersection is the empty set).

[2] **(a)** Draw the graph A_1.

[3] **(b)** Determine the number of vertices and the number of edges in A_n, $n \geq 1$.

[4] **(c)** Determine all values of $n \geq 1$ for which A_n has cycles of length 3.

[5] **(d)** Determine all values of $n \geq 1$ for which A_n is planar.

[10] **5.** Consider the graphs G_1, G_2, and G_3 drawn below. Determine which, if any, of these graphs are isomorphic. If a pair of graphs is isomorphic, give an isomorphism; if a pair of graphs is not isomorphic, prove that they are not.

[7] **6(a)** Prove that if G is a graph on $p \geq 1$ vertices, in which every vertex has degree $\geq \frac{p}{2}$, then G must be connected.

[3] **(b)** For each $p \geq 4$, give an example of a graph on p vertices that is not connected, in which exactly one vertex has degree $< \frac{p}{2}$.

[7] **7(a)** Construct a breadth-first search tree for the graph H below, using vertex labelled 1 as the root vertex. When considering the vertices adjacent to the vertex being examined, add them to the tree in increasing order of label. Give a list of the vertices in the order that they join the tree.

[3] **(b)** Use the breadth-first search tree from (a) to determine whether H is bipartite or not. If H is bipartite, find a bipartition; if H is not bipartite, find an odd cycle.

[2] **8(a)** State Euler's formula for a planar embedding of a connected graph.

[4] **(b)** Prove that a connected planar embedding with all faces of degree 4, and all vertices of degree either 2 or 3, can have at most $s = 6$ faces.

[2] **(c)** Draw a connected planar embedding with all faces of degree 4, and all vertices of degree either 2 or 3, with exactly $s = 6$ faces.

[8] **(d)** Determine whether the graph B below is planar or not.

[10] **9.** Determine a maximum matching and a minimum cover in the graph G below, by applying the maximum matching algorithm, beginning with the matching indicated by the tripled edges in G. You may find the extra drawings of G helpful in any iterations of the matching algorithm that are required.