PLEASE HAND IN

UNIVERSITY OF TORONTO
Faculty of Arts and Science

APRIL 2015 EXAMINATIONS

CSC 373 H1S

Duration—3 hours

PLEASE HAND IN

Examination Aids: One *double*-sided *hand*written *8.5"×11"* aid sheet.

Student Number: $\lfloor\_\_\_\_\_\_\_\_\_\_\rfloor$

Last (Family) Name(s): ______________________________

First (Given) Name(s): ______________________________

---

*Do **not** turn this page until you have received the signal to start.*
In the meantime, please read the instructions below *carefully*.

This final examination consists of 7 questions on 11 pages (including this one), printed on *one* side of the paper. *When you receive the signal to start, please make sure that your copy of the examination is complete and fill in the identification section above.*

Answer each question directly on the examination paper, in the space provided, and use the back of the pages for rough work. If you need more space for one of your solutions, use the back of a page and *indicate clearly the part of your work that should be marked.*

In your answers, you may use without proof any theorem or result covered in lectures, tutorials, problem sets, assignments, or the textbook, as long as you give a clear statement of the result(s)/theorem(s) you are using. You must justify all other facts required for your solutions.

Write up your solutions carefully! In particular, use notation and terminology correctly and explain what you are trying to do —part marks *will* be given for showing that you know the general structure of an answer, even if your solution is incomplete.

If you are unable to answer a question (or part of a question), remember that you will get 20% of the marks for any solution that you leave *entirely blank* (or where you cross off everything you wrote to make it clear that it should not be marked).

*Remember that, in order to pass the course, you must obtain a grade of **at least 40%** (21.6/54) on this final examination.*

MARKING GUIDE

Nº 1:	____/ 5
Nº 2:	____/11
Nº 3:	____/ 6
Nº 4:	____/ 6
Nº 5:	____/10
Nº 6:	____/ 8
Nº 7:	____/ 8
BONUS:	____/ 3
TOTAL:	____/54

Good Luck!

Question 1. *(Greedy Algorithms)* [5 MARKS]

Consider the following algorithm that takes as input an undirected graph G with integer weights $w(v)$ for each **vertex** v, and that attempts to construct a clique in G with the maximum total weight. For this problem, assume that every vertex weight $w(v)$ is a power of 2 and that no two weights are the same.

```
HEAVYCLIQUE(G = (V,E), w : V → Z):
    sort V so that w(v_1) ≥ w(v_2) ≥ ··· ≥ w(v_n)
    C ← ∅
    for i ← 1,2,...,n:
        if v_i is connected to every vertex in C:   # vacuously TRUE when C = ∅
            C ← C ∪ {v_i}
    return C
```

Part (a) [2 MARKS]

Consider the sequence of partial solutions $C_0, C_1, \ldots, C_n$ generated by algorithm HEAVYCLIQUE. List the *cases* in the proof by induction that every partial solution is promising. *There is **nothing to prove** for this question!*

Part (b) [3 MARKS]

For each case you identified in the previous part, list the *sub-cases* (if any) in the proof by induction that every partial solution is promising. *There is **nothing to prove** for this question!*

Question 1. (CONTINUED)

(Bonus) [3 MARKS]

WARNING! This question is long and/or difficult and will be marked *harshly*: credit will be given only for making *significant* progress toward a correct answer. Please attempt this only *after* you have completed the rest of the final examination.

Prove the correctness of algorithm HEAVYCLIQUE, under the assumptions given before the algorithm.

Question 2. *(Dynamic Programming)* [11 MARKS]

A subsequence $(a_{i_1}, a_{i_2}, \ldots, a_{i_k})$ of a sequence of real numbers $(a_1, a_2, \ldots, a_n)$ is *palindromic* when $a_{i_1} = a_{i_k}$, $a_{i_2} = a_{i_{k-1}}$, ..., $a_{i_k} = a_{i_1}$. For example, $(-4, \sqrt{2}, -4)$ is a palindromic subsequence of $(2.5, -4, 17, 17, \sqrt{2}, \pi, -4)$, with $i_1 = 2$, $i_2 = 5$, $i_3 = 7$.

The Longest Palindromic Subsequence ("LPS") optimization problem is defined as follows.

Input: A sequence of real numbers $a_1, a_2, \ldots, a_n$.

Output: A palindromic subsequence $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$ with maximum length k.

In our example, $(-4, 17, 17, -4) = (a_2, a_3, a_4, a_7)$ is a longest palindromic subsequence.

Part (a) [3 MARKS]

Define a *two-dimensional* array to solve this problem using dynamic programming. State clearly the range of valid indices for your array, the sub-problem corresponding to each array entry, and the value stored in each entry.

Part (b) [5 MARKS]

Give a recurrence relation for your array values. Include appropriate base case(s). Justify your answer by discussing the recursive structure of the problem.

Question 2. (CONTINUED)

Part (c) [3 MARKS]

Assume that your array values have already been computed for input $a_1, a_2, \ldots, a_n$. Write an algorithm that ouputs a longest palindromic subsequence of $a_1, a_2, \ldots, a_n$. (You should not need all the space on this page!)

Question 3. *(Network Flow)* [6 MARKS]

Solve the following problem using Network Flow techniques. Describe clearly how to construct an appropriate network, how to reconstruct a solution to this problem from a maximum flow (or a minimum cut) in your network, and why your solution is correct.

Input: A directed graph $G = (V, E)$ representing geographical areas, a subset of vertices $X \subseteq V$ representing *populated* areas, and a subset of vertices $Y \subseteq V$ representing *safe* areas.

Output: A collection of *evacuation routes*: paths in G from each node in X to some node in Y (different nodes from X can end up at the same node from Y) such that no two paths share an edge. If this is not possible, the special value NIL.

Question 4. *(Linear Programming)* [6 MARKS]

Write a Linear Program to solve the following problem. Describe clearly each part of your linear program and what characteristic of the problem it corresponds to.

A cargo plane can carry a maximum *weight* of 100 tons and a maximum *volume* of 60 cubic meters. Three types of cargo can be transported:

- cargo 1 has density 2 tons per cubic meter and revenue \$1,000 per cubic meter;
- cargo 2 has density 1 ton per cubic meter and revenue \$1,200 per cubic meter;
- cargo 3 has density 3 tons per cubic meter and revenue \$8,000 per cubic meter.

Decide how much of each cargo to load onto the plane to maximize the revenue, if you are given 40 cubic meters of cargo 1, 30 cubic meters of cargo 2, and 20 cubic meters of cargo 3.

Question 5. *(NP-Completeness)* [10 MARKS]

Write a *detailed* proof that the following Travelling Salesman Decision problem ("TSD") is *NP*-complete.

Input: An integer bound B and a directed graph $G = (V, E)$ with integer edge weights $w(e)$ for all $e \in E$.

Output: Is there some Hamiltonian cycle in G with total weight *less than or equal to* B?

Question 6. *(Self-Reducibility)* [8 MARKS]

Write a *detailed* proof that the following Travelling Salesman Optimization problem ("TSO") is polytime self-reducible.

Input: A directed graph $G = (V, E)$ with integer edge weights $w(e)$ for all $e \in E$.

Output: A Hamiltonian cycle in G with minimum total weight.

Question 7. *(Approximation Algorithms)* [8 MARKS]

Consider the following Maximum Independent Set ("MIS") optimization problem.

Input: An undirected graph $G = (V, E)$.

Output: An independent set $I \subseteq V$ with maximum size (recall that an independent set I has the property that G contains **no** edge between any two vertices in I).

Part (a) [3 MARKS]

Write a greedy algorithm with input $G = (V, E)$ that attempts to find a large independent set in G. (HINT: Keep your algorithm simple! There is a natural greedy strategy for this problem: use it and don't try anything fancy. Also, you should have a lot more room on this page than what you need for your answer.)

Question 7. (CONTINUED)

Part (b) [3 MARKS]

Suppose that the input graph G has maximum degree 4 (in other words, every vertex in G is connected to *at most* 4 neighbours). What does this imply for the size of the independent set returned by your algorithm, when compared to the total number of vertices in G? Explain.

Part (c) [2 MARKS]

What is the approximation ratio for your algorithm when the input graph has maximum degree 4? Justify.