PLEASE HAND IN

UNIVERSITY OF TORONTO
Faculty of Arts and Science

DECEMBER 2015 EXAMINATIONS

CSC 373 H1F

Duration — 3 hours

PLEASE HAND IN

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

Student Number: ___________

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 15 pages (including this one), printed on both sides 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 a "blank" page for rough work. If you need more space for one of your solutions, use one of the "blank" pages 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, 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), you can get trong20% of the marks by leaving your answer *entirely blank* (or crossing off everything you wrote to make it clear that it should not be marked).

Remember that, in order to pass the course, you must achieve a grade of at least 40% on this final examination.

MARKING GUIDE

Nº 1:	______/ 5
Nº 2:	______/ 5
Nº 3:	______/10
Nº 4:	______/10
Nº 5:	______/10
Nº 6:	______/10
Nº 7:	______/10
BONUS:	______/ 5
TOTAL:	______/60

Good Luck!

Question 1. [5 MARKS]

Consider a small part of the U of T optical network between seven departments, labelled A, B, C, D, E, F, G. The following pairs of departments have a direct fiber-optic connection with the capacity indicated.

$A \to B$: 40 GB/s	$A \to C$: 20 GB/s	$A \to D$: 5 GB/s	$B \to D$: 18 GB/s	$B \to E$: 5 GB/s
$C \to D$: 10 GB/s	$C \to E$: 15 GB/s	$C \to G$: 8 GB/s	$D \to E$: 20 GB/s	$D \to F$: 5 GB/s
$E \to F$: 20 GB/s	$E \to G$: 15 GB/s	$F \to G$: 30 GB/s		

Assume that departments A, B are on Spadina Avenue, departments C, D, E are on St. George Street, and departments F, G are on University Avenue. We wish to know the maximum total bandwidth from departments on Spadina Avenue to departments on University Avenue.
Describe clearly how to model this problem as a network flow problem.

Bonus. [5 MARKS]

For the same problem as above, suppose that department E has a limit of 70 GB/s on the bandwidth it can process. Describe how to modify your answer to account for this new requirement. Justify briefly that your modification is correct.

Question 2. [5 MARKS]

Let $G = (V, E)$ be an undirected graph with $n = |V|$ vertices. We wish to colour every vertex either red, blue, or yellow so that **no** edge has both endpoints with the same colour.

Formulate an integer linear program that is feasible if and only if such a colouring is possible. Label clearly each element of your integer program, then provide a brief justification that it is correct.
(HINT: Use $3n$ variables with 0–1 values. This is slightly different from other problems we have seen because there is no objective function to maximize or minimize. Focus on providing the other elements of a linear program and justifying how solutions to both problems correspond to each other.)

Question 3. [10 MARKS]

Write a detailed proof that the following DISJOINTCLIQUES decision problem is NP-complete.

- **In:** Undirected graph G, positive integer k.
- **Out:** Does G contain at least two *disjoint* cliques of size k? (Two cliques are disjoint if they have no vertex in common.)

Your proof will be marked on its structure as well as its content so explain what you are doing.

Question 4. [10 MARKS]

Consider the following optimization problems.

LONGESTPATH:

- **In:** Undirected graph G, vertices s, t.
- **Out:** A simple path in G from s to t with the maximum number of edges.

LONGESTCYCLE:

- **In:** Undirected graph G.
- **Out:** A simple cycle in G with the maximum number of edges.

Part (a) [6 MARKS]

Show that if LONGESTCYCLE $\in P$, then LONGESTPATH $\in P$. Explain what you are doing and why.
(HINT: Take the time to think carefully about what it means to say that a problem belongs to P.)

Question 4. (CONTINUED)

Part (b) [4 MARKS]

Given your reduction from the previous part and the fact that LONGESTPATH is *NP*-hard, state whether each of the following statements is TRUE or FALSE and *briefly* justify your answers.

i– If there is a $\mathcal{O}(n^3)$ algorithm for LONGESTPATH, then there is a $\mathcal{O}(n^3)$ algorithm for LONGESTCYCLE.

ii– If there is a $\mathcal{O}(n^3)$ algorithm for LONGESTCYCLE, then there is a $\mathcal{O}(n^3)$ algorithm for LONGESTPATH.

iii– If there is a $\mathcal{O}(n^3)$ algorithm for LONGESTCYCLE, then $P = NP$.

Question 5. [10 MARKS]

Suppose you have a weighted undirected graph $G = (V, E)$ and a minimum spanning tree $T \subseteq E$ for G. Consider a new graph G_0 created from G by adding one new vertex v_0 and connecting it with new edges to $k < |V|$ vertices from V.

Part (a) [4 MARKS]

Write an efficient algorithm to find a minimum spanning tree in G_0, given inputs G, T, v_0, and the new edges containing v_0 (with their weights). Include comments or a brief English description.

Part (b) [1 MARK]

What is the time complexity of your algorithm?

Question 5. (CONTINUED)

Part (c) [5 MARKS]

Prove that your algorithm is correct. (HINT: Think carefully about what is required: don't fall into the trap of blindly following a "template" without understanding whether or not it is necessary, and remember to make use of known results.)

Question 6. [10 MARKS]

Consider the following two-player game: start with a list of positive numbers $a_1, a_2, \ldots, a_n$ where n is even; taking turns, each player selects either the current first or current last number until every number has been selected; the player with the largest sum at the end wins.

Part (a) [2 MARKS]

Describe a greedy strategy for playing the game. Then, give a concrete example to show that your strategy does not guarantee that the first player achieves the maximum sum possible.
(HINT: Pick a simple, obvious strategy; there is a short counter-example.)

Part (b) [8 MARKS]

Give a dynamic programming algorithm to determine the maximum sum possible for the first player. To simplify the problem, do **not** make any assumption about how the second player plays: we want to know the maximum possible sum under all possible conditions, not only when the second player plays optimally.

Follow the structure from class to describe your solution. In particular, include a *brief* justification for your algorithm's correctness based on the recursive structure of the problem.
(HINT: In your recurrence, take into account every possible move of the second player.)

Question 6. (CONTINUED)

Part (b) (CONTINUED)

Question 7. [10 MARKS]

Consider the following algorithm to solve the problem from Question 6 (remember that n is even).

$A(a_1, \ldots, a_n)$:
  let $T_1 = a_1 + a_3 + \cdots + a_{n-1}$
  let $T_2 = a_2 + a_4 + \cdots + a_n$
  return $\max(T_1, T_2)$

Part (a) [3 MARKS]

For an arbitrary input $a_1, \ldots, a_n$, let $\text{OPT}(a_1, \ldots, a_n)$ represent the maximum possible sum for the first player, and let $A(a_1, \ldots, a_n)$ represent the value returned by the algorithm State clearly what it means for the algorithm to have approximation ratio **exactly equal to** $r(n)$.

Part (b) [7 MARKS]

State a bound on the approximation ratio of the algorithm above and prove it.
(HINT: Let $T(a_1, \ldots, a_n) = a_1 + \cdots + a_n$.)

Question 7. (CONTINUED)

Part (b) (CONTINUED)

Use the space on this "blank" page for scratch work, or for any solution that did not fit elsewhere.
Clearly label each such solution with the appropriate question and part number.

Use the space on this "blank" page for scratch work, or for any solution that did not fit elsewhere.
Clearly label each such solution with the appropriate question and part number.

Total Marks = 60