CS/ECE 374 A ✧ Spring 2018

Final Exam ♫

May 8, 2018

Real name:	
NetID:	

Gradescope name:	
Gradescope email:	

- ***Don't panic!***
- If you brought anything except your writing implements and your two double-sided 8½" × 11" cheat sheets, please put it away for the duration of the exam. In particular, please turn off and put away *all* medically unnecessary electronic devices.
- Please clearly print your real name, your university NetID, your Gradescope name, and your Gradescope email address in the boxes above. **We will not scan this page into Gradescope.**
- Please also print **only the name you are using on Gradescope** at the top of every page of the answer booklet, except this cover page. These are the pages we will scan into Gradescope.
- Please do not write outside the black boxes on each page; these indicate the area of the page that the scanner can actually see.
- **Please read the entire exam before writing anything.** Please ask for clarification if any question is unclear.
- **The exam lasts 180 minutes.**
- If you run out of space for an answer, continue on the back of the page, or on the blank pages at the end of this booklet, **but please tell us where to look.** Alternatively, feel free to tear out the blank pages and use them as scratch paper.
- As usual, answering any (sub)problem with "I don't know" (and nothing else) is worth 25% partial credit. **Yes, even for problem 1.** Correct, complete, but suboptimal solutions are *always* worth more than 25%. A blank answer is not the same as "I don't know".
- **Please return your cheat sheets and all scratch paper with your answer booklet.**
- ***Good luck!*** And thanks for a great semester!

Beware of the man who works hard to learn something,
learns it, and finds himself no wiser than before.

He is full of murderous resentment of people who are ignorant
without having come by their ignorance the hard way.

— Bokonon

CS/ECE 374 A ✧ Spring 2018 Final Exam ♫ Problem 1	Gradescope name:

For each of the following questions, indicate ***every*** correct answer by marking the “Yes” box, and indicate ***every*** incorrect answer by marking the “No” box. **Assume P ≠ NP.** If there is any other ambiguity or uncertainty, mark the “No” box. For example:

~~Yes~~	No	$2+2=4$
Yes	~~No~~	$x+y=5$
Yes	~~No~~	3SAT can be solved in polynomial time.
~~Yes~~	No	Jeff is not the Queen of England.
~~Yes~~	No	If $P=NP$ then Jeff is the Queen of England.

There are 40 yes/no choices altogether. Each correct choice is worth +½ point; each incorrect choice is worth −¼ point. TO indicate “I don’t know”, write IDK to the left of the Yes/No boxes; each IDK is worth +1/8 point.

(a) Which of the following statements is true for ***every*** language $L \subseteq \{0, 1\}^*$?

Yes	No	L is infinite.
Yes	No	L^* contains the empty string ε.
Yes	No	L^* is decidable.
Yes	No	If L is regular then $(L^*)^*$ is regular.
Yes	No	If L is the intersection of two decidable languages, then L is decidable.
Yes	No	If L is the intersection of two undecidable languages, then L is undecidable.
Yes	No	If L is the complement of a regular language, then L^* is regular.
Yes	No	If L has an infinite fooling set, then L is undecidable.
Yes	No	L is decidable if and only if its complement $\overline{L}$ is undecidable.

(b) Which of the following statements is true for ***every*** directed graph $G = (V, E)$?

Yes	No	$E \neq \varnothing$.
Yes	No	Given the graph G as input, Floyd-Warshall runs in $O(E^3)$ time.
Yes	No	If G has at least one source and at least one sink, then G is a dag.
Yes	No	We can compute a spanning tree of G using whatever-first search.
Yes	No	If the edges of G are weighted, we can compute the shortest path from any node s to any node t in $O(E \log V)$ time using Dijkstra's algorithm.

(c) Which of the following languages over the alphabet $\{\texttt{0}, \texttt{1}\}$ are ***regular***?

Yes	No	$\{\texttt{0}^m\texttt{1}\texttt{0}^n \mid m \leq n\}$
Yes	No	$\{\texttt{0}^m\texttt{1}\texttt{0}^n \mid m + n \geq 374\}$
Yes	No	Binary representations of all perfect squares
Yes	No	$\{xy \mid yx \text{ is a palindrome}\}$
Yes	No	$\{\langle M \rangle \mid M \text{ accepts a finite number of non-palindromes}\}$

(d) Which of the following languages are ***decidable***?

Yes	No	Binary representations of all perfect squares
Yes	No	$\{xy \in \{\texttt{0}, \texttt{1}\}^* \mid yx \text{ is a palindrome}\}$
Yes	No	$\{\langle M \rangle \mid M \text{ accepts the binary representation of every perfect square}\}$
Yes	No	$\{\langle M \rangle \mid M \text{ accepts a finite number of non-palindromes}\}$
Yes	No	The set of all regular expressions that represent the language $\{\texttt{0}, \texttt{1}\}^*$. (This is a language over the alphabet $\{\texttt{Ø}, \texttt{ε}, \texttt{0}, \texttt{1}, \star, +, (,)\}$.)

(e) Which of the following languages can be proved undecidable ***using Rice's Theorem***?

Yes	No	$\{\langle M\rangle \mid M \text{ accepts a finite number of strings}\}$
Yes	No	$\{\langle M\rangle \mid M \text{ accepts both } \langle M\rangle \text{ and } \langle M\rangle^R\}$
Yes	No	$\{\langle M\rangle \mid M \text{ accepts exactly 374 palindromes}\}$
Yes	No	$\{\langle M\rangle \mid M \text{ accepts some string } w \text{ after at most } \lvert w\rvert^2 \text{ steps}\}$

(f) Suppose we want to prove that the following language is undecidable.

$$\textsc{Chalmers} := \{\langle M\rangle \mid M \text{ accepts both } \texttt{STEAMED} \text{ and } \texttt{HAMS}\}$$

Professor Skinner suggests a reduction from the standard halting language

$$\textsc{Halt} := \{\langle M\rangle\#w \mid M \text{ halts on inputs } w\}.$$

Specifically, suppose there is a Turing machine Ch that decides $\textsc{Chalmers}$. Professor Skinner claims that the following algorithm decides $\textsc{Halt}$.

```
DecideHalt(⟨M⟩#w):
  Encode the following Turing machine:
      AuroraBorealis(x):
        if x = STEAMED or x = HAMS or x = ALBANY
            run M on input w
            return False
        else
            return True
  return Ch(⟨AuroraBorealis⟩)
```

Which of the following statements is true for all inputs $\langle M\rangle\#w$?

Yes	No	If M accepts w, then $\textsc{AuroraBorealis}$ accepts $\texttt{CLAMS}$.
Yes	No	If M rejects w, then $\textsc{AuroraBorealis}$ rejects $\texttt{UTICA}$.
Yes	No	If M rejects w, then $\textsc{AuroraBorealis}$ halts on every input string.
Yes	No	If M accepts w, then Ch accepts $\langle\textsc{AuroraBorealis}\rangle$.
Yes	No	$\textsc{DecideHalt}$ decides the language $\textsc{Halt}$. (That is, Professor Skinner's reduction is actually correct.)
Yes	No	$\textsc{DecideHalt}$ actually runs (or simulates) M.
Yes	No	We could have proved $\textsc{Chalmers}$ is undecidable using Rice's theorem instead of this reduction.

(g) Consider the following pair of languages:

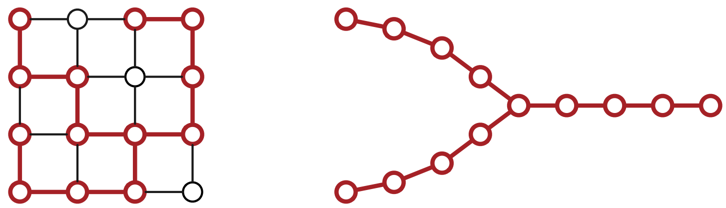
- $\textsc{3Color} := \{G \mid G \text{ is a 3-colorable undirected graph}\}$
- $\textsc{Tree} := \{G \mid G \text{ is a connected acyclic undirected graph}\}$

(For concreteness, assume that in both of these languages, graphs are represented by adjacency matrices.) Which of the following ***must*** be true, assuming P≠NP?

Yes	No	$\textsc{Tree} \cup \textsc{3Color}$ is NP-hard.
Yes	No	$\textsc{Tree} \cap \textsc{3Color}$ is NP-hard.
Yes	No	$\textsc{3Color}$ is undecidable.
Yes	No	There is a polynomial-time reduction from $\textsc{3Color}$ to $\textsc{Tree}$.
Yes	No	There is a polynomial-time reduction from $\textsc{Tree}$ to $\textsc{3Color}$.

CS/ECE 374 A ✧ Spring 2018 Final Exam ♫ Problem 2	Gradescope name:

A ***wye*** is an undirected graph that looks like the capital letter Y. More formally, a wye consists of three paths of equal length with one common endpoint, called the *hub*.



This grid graph contains a wye whose paths have length 4.

Prove that the following problem is NP-hard: Given an undirected graph G, what is the largest wye that is a subgraph of G? The three paths of the wye must not share any vertices except the hub, and they must have exactly the same length.

CS/ECE 374 A ✧ Spring 2018 Final Exam ♫ Problem 3	Gradescope name:

Fix the alphabet $\Sigma = \{0, 1\}$. Recall that a *run* in a string $w \in \Sigma^*$ is a maximal non-empty substring in which all symbols are equal. For example, the string `000001000111111101` consists of exactly six runs: $\texttt{00000100011111110} = \texttt{00000} \bullet \texttt{1} \bullet \texttt{000} \bullet \texttt{1111111} \bullet \texttt{0} \bullet \texttt{1}$.

(a) Let L be the set of all strings in Σ^* where every run has odd length. For example, L contains the string `000100000`, but L does not contain the string `00011`.

Describe both a regular expression for L and a DFA that accepts L.

(b) Let L' be the set of all strings in Σ^* that have the same number of even-length runs and odd-length runs. For example, L' does not contain the string `000011101`, because it has three odd-length runs but only one even-length run, but L' does contain the string `0000111011`, because it has two runs of each parity.

Prove that L' is not regular.

Suppose we want to split an array $A[1..n]$ of integers into k contiguous intervals that partition the sum of the values as evenly as possible. Specifically, define the *cost* of such a partition as the maximum, over all k intervals, of the sum of the values in that interval; our goal is to minimize this cost. Describe and analyze an algorithm to compute the minimum cost of a partition of A into k intervals, given the array A and the integer k as input.

For example, given the array $A = [1, 6, -1, 8, 0, 3, 3, 9, 8, 8, 7, 4, 9, 8, 9, 4, 8, 4, 8, 2]$ and the integer $k = 3$ as input, your algorithm should return the integer 37, which is the cost of the following partition:

$$\Big[\ \overbrace{1, 6, -1, 8, 0, 3, 3, 9, 8}^{37}\ \Big|\ \overbrace{8, 7, 4, 9, 8}^{36}\ \Big|\ \overbrace{9, 4, 8, 4, 8, 2}^{35}\ \Big]$$

The numbers above each interval show the sum of the values in that interval.

CS/ECE 374 A ✧ Spring 2018 Final Exam ♫ Problem 5	Gradescope name:

(a) Fix the alphabet $\Sigma = \{\texttt{0}, \texttt{1}\}$. Describe and analyze an efficient algorithm for the following problem: Given an NFA M over Σ, does M accept at least one string? Equivalently, is $L(M) \neq \varnothing$?

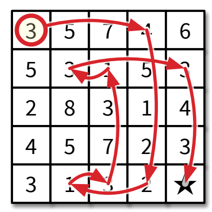
(b) Recall from Homework 10 that deciding whether a given NFA accepts *every* string is NP-hard. Also recall that the complement of every regular language is regular; thus, for any NFA M, there is another NFA M' such that $L(M') = \Sigma^* \setminus L(M)$. So why doesn't your algorithm from part (a) imply that P=NP?

CS/ECE 374 A ✧ Spring 2018 Final Exam ♫ Problem 6	Gradescope name:

A ***number maze*** is an $n \times n$ grid of positive integers. A token starts in the upper left corner; your goal is to move the token to the lower-right corner. On each turn, you are allowed to move the token up, down, left, or right; the distance you may move the token is determined by the number on its current square. For example, if the token is on a square labeled 3, then you may move the token three steps up, three steps down, three steps left, or three steps right. However, you are never allowed to move the token off the edge of the board.

Describe and analyze an efficient algorithm that either returns the minimum number of moves required to solve a given number maze, or correctly reports that the maze has no solution.

3	5	7	4	6
5	3	1	5	3
2	8	3	1	4
4	5	7	2	3
3	1	3	2	★



A 5 × 5 number maze that can be solved in eight moves.

(scratch paper)

(scratch paper)

(scratch paper)

(scratch paper)

Some useful NP-hard problems. You are welcome to use any of these in your own NP-hardness proofs, except of course for the specific problem you are trying to prove NP-hard.

CircuitSat: Given a boolean circuit, are there any input values that make the circuit output True?

3Sat: Given a boolean formula in conjunctive normal form, with exactly three distinct literals per clause, does the formula have a satisfying assignment?

MaxIndependentSet: Given an undirected graph G, what is the size of the largest subset of vertices in G that have no edges among them?

MaxClique: Given an undirected graph G, what is the size of the largest complete subgraph of G?

MinVertexCover: Given an undirected graph G, what is the size of the smallest subset of vertices that touch every edge in G?

MinSetCover: Given a collection of subsets $S_1, S_2, \ldots, S_m$ of a set S, what is the size of the smallest subcollection whose union is S?

MinHittingSet: Given a collection of subsets $S_1, S_2, \ldots, S_m$ of a set S, what is the size of the smallest subset of S that intersects every subset S_i?

3Color: Given an undirected graph G, can its vertices be colored with three colors, so that every edge touches vertices with two different colors?

HamiltonianPath: Given graph G (either directed or undirected), is there a path in G that visits every vertex exactly once?

HamiltonianCycle: Given a graph G (either directed or undirected), is there a cycle in G that visits every vertex exactly once?

TravelingSalesman: Given a graph G (either directed or undirected) with weighted edges, what is the minimum total weight of any Hamiltonian path/cycle in G?

LongestPath: Given a graph G (either directed or undirected, possibly with weighted edges), what is the length of the longest simple path in G?

SteinerTree: Given an undirected graph G with some of the vertices marked, what is the minimum number of edges in a subtree of G that contains every marked vertex?

SubsetSum: Given a set X of positive integers and an integer k, does X have a subset whose elements sum to k?

Partition: Given a set X of positive integers, can X be partitioned into two subsets with the same sum?

3Partition: Given a set X of $3n$ positive integers, can X be partitioned into n three-element subsets, all with the same sum?

IntegerLinearProgramming: Given a matrix $A \in \mathbb{Z}^{n \times d}$ and two vectors $b \in \mathbb{Z}^n$ and $c \in Z^d$, compute $\max\{c \cdot x \mid Ax \leq b, x \geq 0, x \in \mathbb{Z}^d\}$.

FeasibleILP: Given a matrix $A \in \mathbb{Z}^{n \times d}$ and a vector $b \in \mathbb{Z}^n$, determine whether the set of feasible integer points $\max\{x \in \mathbb{Z}^d \mid Ax \leq b, x \geq 0\}$ is empty.

Draughts: Given an $n \times n$ international draughts configuration, what is the largest number of pieces that can (and therefore must) be captured in a single move?

SuperMarioBrothers: Given an $n \times n$ Super Mario Brothers level, can Mario reach the castle?

SteamedHams: Aurora borealis? At this time of year, at this time of day, in this part of the country, localized entirely within your kitchen? May I see it?