PRACTICE Exam 2

Name: ______________________________

NOTES:

- This is a practice exam. It will not be graded. The aim is to give you a sense of the types of questions that may appear on the exam.
- As a reminder, this is not an exhaustive list of types of problems, anything through Thursday April 3 is fair game.
- I recommend preparing your notes sheet before attempting these problems, then revising it to see if it needs to be adapted.
- Attempt all problems. On the exam, it is possible to receive partial credit for showing your work.

Reference Page

This page does not have any questions. Instead it is a reference for relevant definitions and theorems for this exam.

P is the class of languages which are decidable in polynomial time on a deterministic Turing machine.

NP is the class of languages which have polynomial time verifiers.

NP-complete is the class of languages which are both in NP and every language in NP is polynomial-time reducible to it.

$A \leq_p B$ means that A is polynomial-time reducible to B. In particular, there is a polynomial-time computable function $f : \Sigma^* \to \Sigma^*$, where for every w,

$$w \in A \text{ if and only if } f(w) \in B.$$

PSPACE is the class of languages which are decidable in polynomial space on a deterministic Turing machine.

PSPACE-complete is the class of languages which are in PSPACE and for which every language in PSPACE is polynomial-time reducible to it.

L is the class of languages which are decidable in logarithmic space on a deterministic Turing machine.

NL is the class of languages which are decidable in logarithmic space on a nondeterministic Turing machine.

Savitch's theorem states that for any function $F : \mathbb{N} \to \mathbb{R}^+$, where $f(n) \geq n$,

$$\mathsf{NSPACE}(f(n)) \subseteq \mathsf{SPACE}(f^2(n)).$$

Problem 1	Score: /

Consider a multiset (a set that allows for repeated elements) S. The *3-partition* problem involves partitioning the elements of S into triples, *i.e.*, (x, y, z) with $x, y, z \in S$, such that every triple sums to the same value. For example,

$$\{ 0, 0, 0, 2, 2, 2, 3, 3, 6 \}$$

can be partitioned into the triples $(0, 0, 6)$, $(3, 3, 0)$, and $(2, 2, 2)$, all of which sum to 6.
Show that the 3-partition problem is in NP.

Problem 2	Score: /

Recall the following problems over undirected graphs:
Hamiltonian path:

$$\text{HAMPATH} = \{ \langle G, x, y\rangle \mid \text{There is a path from } x \text{ to } y \text{ in } G \text{ that visits every node exactly once} \}.$$

Longest paths:

$$\text{LPATH} = \{ \langle G, x, y, t\rangle \mid \text{There is a path from } x \text{ to } y \text{ in } G \text{ of length at least } t \}.$$

Come up with a reduction that would show

$$\text{HAMPATH} \leq_p \text{LPATH}.$$

You only need to give the mapping, you do not need to prove that it is correct.
(*Hint*: what is the relationship between the length of a Hamiltonian path and the number of vertices in G?)
(*note:* A different version of this exam problem might present to you the reduction, and ask you to prove that it is correct.)

Problem 3	Score: /

Consider the following two variants of problems involving finding paths between points in a simple, (non-negative) weighted graph:

$$\mathsf{SPATH} = \{\, \langle G, x, y, t \rangle \mid \text{There is a path from } x \text{ to } t \text{ in } G \text{ of length at most } t \,\}$$

$$\mathsf{LPATH} = \{\, \langle G, x, y, t \rangle \mid \text{There is a path from } x \text{ to } t \text{ in } G \text{ of length at least } t \,\}$$

It turns out that SPATH is in P and LPATH is NP-complete!
Suppose you are at a logistics company, writing software to route cargo between cities. You are given a choice of two different assignments:

1. Write software to find the shortest path to ship a given piece of freight from one city to another.
2. Write software to find a path through the country from one city to another that maximizes deliveries.

Which assignment would you choose? Explain your answer in a few sentences.
(*Note*: there is not a *definite* right answer here. I'm interested in your choice coupled with your reasoning based on the lessons of this course.)

Problem 4	Score: /

Consider the *vertex cover* problem from our discussion of NP-completeness:

$$\mathsf{COVER} = \{\, (G, k) \mid \text{Graph } G \text{ has a vertex cover of size } k \,\}$$

A k-vertex cover of a graph is a k-subset of the vertices that cover all the edges of G.
Show that COVER $\in$ PSPACE.

Problem 5 Score: /

Consider the following claim:

$$\mathsf{NPSPACE} = \mathsf{EXPTIME}.$$

(a) In a sentence or two, describing what this claim is stating in informal terms. Specifically, describe the nature of the problems in each of these complexity classes and what is the relationship between them implied by the claim.

(b) In a few sentences, state whether you believe the claim to be true and the justification for your reasoning.

(*Note*: to be clear, the claim has not yet been proven or disproven! Its status is currently unknown. I want you most-informed opinion of the matter based on what you know of the complexity classes involved.)

Problem 6	Score: /

(a) Observe that a non-deterministic finite automata N is not a "real" machine (modulo the Automata Processor that we studied in a prior problem set). To simulate N, we must apply the subset transformation on N to obtain an equivalent deterministic finite automata D. Based on this, what is the time and space complexity of simulating an NFA?

(b) In general, when we attempt to simulate a non-deterministic machine, what kind of performance blow-up can we expect? In a few sentences, describe where this blow-up comes from.