"CS/ECE 374 A": Algorithms & Models of Computation, Spring 2025

Final Exam — May 15, 2025

Name:	
NetID:	

- Please *clearly PRINT* your name and your NetID in the boxes above.
- This is a closed-book but you are allowed a 2 pages (4 sides) hand written cheat sheet that you have to submit along with your exam. If you brought anything except your writing implements, put it away for the duration of the exam. In particular, you may not use *any* electronic devices.
- **Please read the entire exam before writing anything.** Please ask for clarification if any question is unclear. The exam has 7 problems, each worth 10 points.
- **You have 180 minutes (3 hours) for the exam.**
- 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.**
- **Write everything inside the box around each page.** Anything written outside the box may be cut off by the scanner.
- **Proofs are required only if we specifically ask for them.** You may state and use (without proof or justification) any results proved in class or in the problem sets unless we explicitly ask you for one.
- You can do hard things!
- **Do not cheat.** You know the student code and all that jazz. Grades do matter, but not as much as you may think, and your values are more important.

---

1 Short Answer and True/False

(2.5 pts) Give asymptotically tight bounds for the following sum and recurrence:

(a) $\sum_{i=1}^{n}(\log i)^2$

(b) $T(n) = 2T(n-2) + 1$ for $n > 2$; $T(n) = 1$ for $n \leq 2$.

(7.5 pts) True/False: fill in the box corresponding to your answer.

(c) Every language in NP is decidable.
○ True ○ False

(d) There is a polynomial-time reduction from 10-Color to 3-Color.
○ True ○ False

(e) If L is regular, L^* is in P.
○ True ○ False

(f) Let G be a DAG such that for every pair of vertices u, v, either u can reach v or v can reach u. Then G has a Hamiltonian Path.
○ True ○ False

(g) Let $G = (V, E)$ be a connected graph with distinct edge weights. Then for every vertex v, the MST of G contains the lightest edge that is adjacent to v.
○ True ○ False

(g) The MST of a graph G is unique if and only if its edge weights are distinct.
○ True ○ False

(i) Let G be a graph with non-negative edge weights. Then the *maximum* weight spanning tree G can be computed in polynomial time.
○ True ○ False

(j) Given a graph G where every edge has length 100 or 374, the shortest s-t path can be computed in linear time.
○ True ○ False

(k) If L and its complement $\bar{L}$ are recursively enumerable then L is decidable.
○ True ○ False

(l) If L is recursively enumerable then L^* is recursively enumerable.
○ True ○ False

2 Classification

(10 pts) For each of the problems below choose the *best* running time/complexity from the list below, based on what you know from the course. Fill in a **single circle** corresponding to your choice.

- L There is an algorithm that runs in linear time
- Q There is an algorithm that runs in quadratic time
- C There is an algorithm that runs in cubic time
- P There is a polynomial-time algorithm
- NP The problem is in NP
- D The problem is decidable
- U The problem is undecidable

(a) Given a directed graph G, check whether G is a DAG.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(b) $L = \{\langle M \rangle, w \mid M \text{ rejects } w.\}$
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(c) Checking whether a given array A of n numbers is sorted.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(d) Computing the minimum sized vertex cover in a given tree T.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(e) $L = \{\langle M \rangle, w \mid M \text{ is a TM that accepts } w \text{ in } |w|^4 \text{ steps.}\}$.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(f) Given undirected graph G with n nodes checking if G has an independent set of size at least $n - 10$.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(g) Computing the longest increasing subsequence in a sequence of n numbers.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(h) Given three DFAs M_1, M_2, and M_3, constructing the DFA M that accepts $L(M_1) \cap L(M_2) \cap L(M_3)$ via the product construction.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(i) Checking if G has a spanning tree with at most 2 leaves.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

(j) Finding the longest s-t path in a undirected graph.
○ L ○ Q ○ C ○ P ○ NP ○ D ○ U

3 Shortest Paths

Let $G = (V, E)$ be a directed graph with edge lengths $\ell(e), e \in E$. These lengths could be postive, zero, or negative. Let s, t be two distinct nodes in the graph. Describe an efficient algorithm that outputs the length of the shortest s-t walk in G that contains at most one negative length edge; note that if the walk contains a negative length edge then it can be used only once.

4 NP-Completeness

Given a graph $G = (V, E)$ a cycle cover is a set of cycles $\{C_1, C_2, \ldots, C_h\}$ for some $h \geq 1$ such that each vertex v is contained in at least one of the cycles. Note that the cycles are not required to be vertex or edge disjoint. Prove the problem of deciding whether a given graph G has a cycle cover with at most 3 cycles is NP-Hard.

5 Undecidability

Prove that $L = \{\langle M \rangle \mid M$ accepts *at most* 5 binary strings$\}$ is undecidable. You are not allowed to use Rice's theorem for this question.

6 Work and Wait

You are given a sequence of n tasks. If you do the ith task, you receive a reward of Value$[i]$. However, this will also cause you to be too busy to do the next Wait$[i]$ tasks—that is, you will not be able to perform tasks $i+1$ through $i+\text{Wait}[i]$. Additionally, you can do at most k tasks total before you become too tired to do any more. Describe an algorithm that, on input Value$[1..n]$, Wait$[1..n]$, and k, computes the maximum total reward you can get. You can assume all the numbers are non-negative.

7 Regularity/DFA/NFAs

Let $M_1 = (Q_1, \Sigma, \delta_1, s_1, A_1)$, $M_2 = (Q_2, \Sigma, \delta_2, s_2, A_2)$, and $M_3 = (Q_3, \Sigma, \delta_3, s_3, A_3)$ be DFAs and let L_1, L_2, L_3 be the languages accepted by them.

(a) (5 pts) Describe a DFA M that accepts the language $(L_1 - L_2) \cap L_3$.

(b) (5 pts) Describe an algorithm that given M_1, M_2, M_3 checks whether $(L_1 - L_2) \cap L_3 = \emptyset$.

This page is for extra work.

This page is for extra work.

List of some useful NP-Complete Problems/Languages

- **SAT** $\{\varphi \mid \varphi$ is a boolean formula in CNF form and is satisfiable$\}$
- **3SAT** $\{\varphi \mid \varphi$ is a boolean formula in CNF form with exactly 3 literals per clause and is satisfiable$\}$
- **Circuit-SAT** $\{C \mid C$ is a boolean circuit such that there is a setting of values to the inputs to C that make it evaluate to TRUE$\}$
- **Independent Set** $\{\langle G, k\rangle \mid$ Graph $G = (V, E)$ has a subset of vertices $V' \subseteq V$ of size at least k such that no two vertices in V' are connected by an edge$\}$
- **Vertex Cover** $\{\langle G, k\rangle \mid$ Graph $G = (V, E)$ has a subset of vertices $V' \subseteq V$ of size at most k such that every edge in E has at least one of its endpoints in $V'\}$
- **Clique** $\{\langle G, k\rangle \mid$ Graph G has a complete subgraph of size at least $k\}$
- **3Color** $\{\langle G\rangle \mid$ The vertices of graph G can be colored with 3 colors so that no two adjacent vertices share a color$\}$
- **Coloring** $\{\langle G, k\rangle \mid$ The vertices of graph G can be colored with k colors so that no two adjacent vertices share a color$\}$
- **Hamiltonian Cycle** $\{\langle G\rangle \mid$ Directed graph G contains a directed cycle visiting each vertex exactly once$\}$
- **Undir Hamiltonian Cycle** $\{\langle G\rangle \mid$ Undirected graph G contains a cycle visiting each vertex exactly once$\}$
- **Hamiltonian Path** $\{\langle G\rangle \mid$ Directed graph G contains a directed path visiting each vertex exactly once$\}$
- **Undir Hamiltonian Path** $\{\langle G\rangle \mid$ Undirected graph G contains a path visiting each vertex exactly once$\}$