ECE 374 B: Algorithms and Models of Computation, Fall 2025
Midterm 3 – December 04, 2025

- **You will have 75 minutes (1.25 hours) to solve all the problems. Most have multiple parts.** Don't spend too much time on questions you don't understand and focus on answering as much as you can!
- ***BUDGET YOUR TIME WISELY***. I highly recommend working on the questions you know first and the questions you need to think about second.
- *No* resources are allowed for use during the exam except a multi-page cheatsheet and scratch paper on the back of the exam. ***Do not tear out the cheatsheet or the scratch paper!*** It messes with the auto-scanner.
- You should write your answers *completely* in the space given for the question. We will not grade parts of any answer written outside of the designated space.
- Please *use a dark-colored pen* unless you are *absolutely* sure your pencil writing is forceful enough to be legible when scanned. We reserve the right to take off points if we have difficulty reading the uploaded document.
- Unless otherwise stated, assume $\mathsf{P} \neq \mathsf{NP}$.
- Assume that whenever the word "reduction" is used, we mean a (not necessarily polynomial-time) *mapping/many-one* reduction.
- You can only refer to the cheat sheet content as a black box.
- ***Don't cheat.*** If we catch you, you will get an F in the course.
- ***Good luck!***

Name: ______________________________

NetID: ______________________________

Date: ______________________________

1 Short Answer I (2 questions) - 10 points

For each of the reductions containing a known problem and an unknown problem (X) please circle the complexity classes that X **MAY** belong to. Only circle the classes, no explanation is needed. There will be no partial credit. If no complexity classes should be selected, please circle "none of the above."

(a) **SAT**: Given a conjunctive normal formula, determine if there is a truth assignment that makes the formula evaluate to true.

Reduction: SAT $\leq_P$ X

Choose the classes that X **MAY** belong to:

P NP NP-complete Decidable Undecidable

None of the above

(b) **LIS**: Given a sequence A and an integer k, return TRUE if the longest increasing subsequence is more than k in length. FALSE otherwise.

Reduction: LIS $\leq_P$ X

Choose the classes that X **MAY** belong to:

P NP NP-complete Decidable Undecidable

None of the above

2 Short Answer II (2 questions) - 10 points

For each of the reductions containing a known problem and an unknown problem (X) please circle the complexity classes that X **MUST** belong to. Only circle the classes, no explanation is needed. There will be no partial credit. If no complexity classes should be selected, please circle "none of the above."

(a) **SAT**: Given a conjunctive normal formula, determine if there is a truth assignment that makes the formula evaluate to true.

Reduction: X $\leq_P$ SAT

Choose the classes that X **MUST** belong to:

P NP NP-complete Decidable Undecidable

None of the above

(b) **HALT**: Given a TM M and input string x, if $M(w)$ halts, return TRUE, otherwise return false.

Reduction: X $\leq_P$ HALT

Choose the classes that X **MUST** belong to:

P NP NP-complete Decidable Undecidable

None of the above

3 Short Answer III (4 questions) - 20 points

For each of the problems provide a brief and concise solution. These are short answer questions and partial credit will be limited.

(a) Write an example of a unsatisfiable 3SAT formula where the literals within a clause are distinct.

(b) For the SAT formula snippet below, transform it to a **3**SAT formula snippet. You may not make any assumptions for how the literals are used in the other clauses. If you add any extra variables, clearly state/mark that you did so.

$$\ldots \wedge (w \vee x \vee y \vee z) \wedge (u \vee v) \wedge \ldots$$

(c) Given a directed, **fully-connected** graph ($G = (V, E)$), what is the minimum value of k (minimum number of colors), needed to color this graph.

(d) You know that a problem (X) is context-free. Is it decidable or undecidable?

4 Classification I (P/NP) - 12 points

Is the following problem in P, NP, or some combinations of complexity classes? For each of the following problems, circle all the complexity classes that problem belongs to. Whatever class it is in, prove it!

A **TasteTheRainbow(TTR) coloring** asks if there is a coloring for an undirected graph $G = (V, E)$ where each vertex can be colored such that no two adjacent vertices are colored the same, *and* there are an equal number of vertices for each color.

- INPUT: A undirected graph G and integer k.
- OUTPUT: TRUE if there exists a coloring with at most k colors and all colors are used equally. FALSE otherwise.

Which of the following complexity classes does this problem belong to? Circle ***all*** that apply:

P NP NP-hard NP-complete

5 Classification II (P/NP) - 12 points

Is the following problem in P, NP, or some combinations of complexity classes? For each of the following problems, circle all the complexity classes that problem belongs to. Whatever class it is in, prove it!

A **HamilDAG** problem asks if there is a path that visits every vertex in a directed acyclic graph exactly once.

- INPUT: A directed acyclic graph G
- OUTPUT: TRUE if there exists a path that visits every vertex exactly once. FALSE otherwise.

Which of the following complexity classes does this problem belong to? Circle ***all*** that apply:

P NP NP-hard NP-complete

6 Classification (Decidability) - 12 points

Same type of problem as before, but now you have to determine if it is decidable or not and provide a concise explanation/proof why.

A **AsManyAsPossibleSAT (AMAPSAT)** problem asks if a formula ϕ has a truth assignment that can satisfy atleast k clauses.

- INPUT: A formula ϕ and an integer k
- OUTPUT: TRUE if there exists a truth assignment that satisfies at least k clauses. FALSE otherwise.

Which of the following complexity classes does this problem belong to? Circle ***all*** that apply:

decidable   undecidable

7 Classification (Mixed) I - 12 points

Same type of problem as before, but now you have to determine decidability in addition to algorithmic complexity.

Given the language:

$$\textsc{PayAttention}_{TM} = \{\langle M \rangle \mid M \text{ accepts on strings "ECE374" and "KANI" }\}$$

Which of the following classes does this language belong to? Whatever you choose, *succinctly* prove it!

P NP NP-hard NP-complete decidable undecidable

8 Classification (Mixed) II - 12 points

Same type of problem as before, but now you have to determine decidability in addition to algorithmic complexity.

Given the language:

$$\text{SERIOUSLY}_{DFA} = \{\langle A \rangle \mid A \text{ is a DFA that accepts atleast one string } w \text{ where } |w| \leq 374\}$$

Which of the following classes does this language belong to? Whatever you choose, *succinctly* prove it!

P NP NP-hard NP-complete decidable undecidable

EXTRA CREDIT (1 pt)

Give an example of a problem that is not in NP, but is in EXPTIME time.

EXTRA CREDIT (? pt)

What will your pre-curve grade, not including extra credit, be on this exam? If you guess your integer score, I'll add 20 points to your exam grade. You must guess a integer between [20-100] (yes you must score atleast 20 points to be eligible for this EC). We'll round down your actual score to the nearest integer.

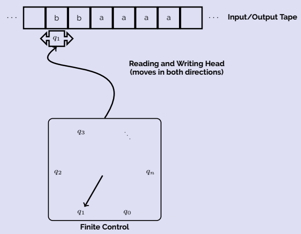
This page is for additional scratch work!

ECE 374 B Reductions, P/NP, and Decidability: Cheatsheet

Turing Machines

Turing machine is the simplest model of computation.

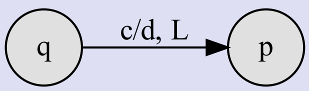
- Input written on (infinite) one sided tape.
- Special blank characters.
- Finite state control (similar to DFA).
- Ever step: Read character under head, write character out, move the head right or left (or stay).
- Every TM **M** can be encoded as a string $\langle M \rangle$



Transition Function: $\delta : Q \times \Gamma \to Q \times \Gamma \times \{\leftarrow, \rightarrow, \square\}$

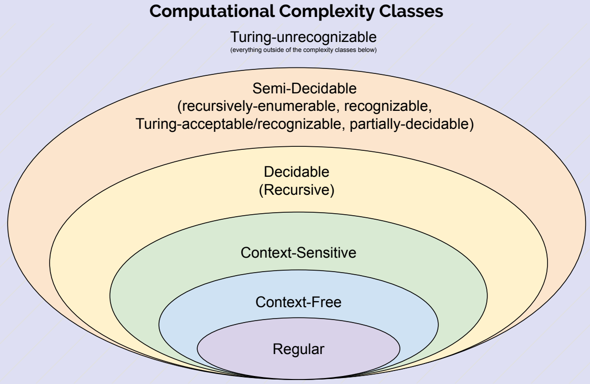
$\delta(q, c) = (p, d, \leftarrow)$

- q: current state.
- c: character under tape head.
- p: new state.
- d: character to write under tape head
- $\leftarrow$: Move tape head left.

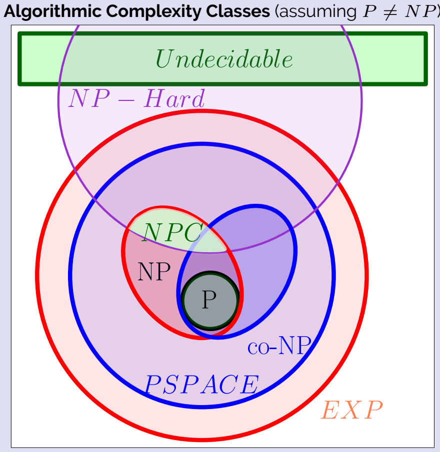


Complexity Classes

Computational Complexity Classes



Algorithmic Complexity Classes (assuming $P \neq NP$)



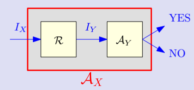
Reductions

A general methodology to prove impossibility results.

- Start with some *known* hard problem X
- *Reduce* X to your favorite problem Y

If Y can be solved then so can $X \implies Y$. But we know X is hard so Y has to be hard too. On the other hand if we know Y is easy, then X has to be easy too.

The Karp reduction, $X \leq_P Y$ suggests that there is a polynomial time reduction from X to Y.



Assuming

- $R(n)$: running time of $\mathcal{R}$
- $Q(n)$: running time of $\mathcal{A}_Y$

Running time of $\mathcal{A}_X$ is $O(Q(R(n)))$

Sample NP-complete problems

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?

IndependentSet: Given an undirected graph G and integer k, what is there a subset of vertices $\geq k$ in G that have no edges among them?

Clique: Given an undirected graph G and integer k, is there a complete complete subgraph of G with more than k vertices?

kPartition: Given a set X of kn positive integers and an integer k, can X be partitioned into n, k-element subsets, all with the same sum?

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?

LongestPath: Given a graph G (either directed or undirected, possibly with weighted edges) and an integer k, does G have a path $\geq k$ length?

- Remember a **path** is a sequence of distinct vertices $[v_1, v_2, \ldots v_k]$ such that an edge exists between any two vertices in the sequence. A **cycle** is the same with the addition of a edge $(v_k, v_1) \in E$. A **walk** is a path except the vertices can be repeated.
- A formula is in conjunction normal form if variables are or'ed together inside a clause and then clauses are and'ed together: $((x_1 \vee x_2 \vee x_3) \wedge (\overline{x_2} \vee x_4 \vee x_5))$. Disjunctive normal form is the opposite $((x_1 \wedge x_2 \wedge x_3) \vee (\overline{x_2} \wedge x_4 \wedge x_5))$.

Sample undecidable problems

AcceptOnInput: $A_{TM} = \{\langle M, w \rangle \mid M \text{ is a TM and } M \text{ accepts on } w\}$

HaltsOnInput: $Halt_{TM} = \{\langle M, w \rangle \mid M \text{ is a TM and halts on input } w\}$

HaltOnBlank: $HaltB_{TM} = \{\langle M \rangle \mid M \text{ is a TM \& } M \text{ halts on blank input}\}$

Emptiness: $E_{TM} = \{\langle M \rangle \mid M \text{ is a TM and } L(M) = \emptyset\}$

Equality: $EQ_{TM} = \left\{\langle M_A, M_B \rangle \;\middle|\; \begin{matrix} M_A \text{ and } M_B \text{ are TM's} \\ \text{and } L(M_A) = L(M_B) \end{matrix}\right\}$