CS/ECE 374 A ✧ Spring 2018

Fake Final Exam

May 6, 2018

Real name:	Jeff E
NetID:	jeffe

Gradescope name:	Flash
Gradescope email:	Flash@gordon.mil

- ***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.**
- May the Sith be with you.

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

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	
X		$2+2=4$
	X	$x+y=5$
	X	3SAT can be solved in polynomial time.
X		Jeff is not the Queen of England.
X		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	
	X	L contains the empty string ε.
	X	L^* is infinite.
	X	L^* is regular.
X		L is infinite or L is decidable (or both).
X		If L is the union of two regular languages, then L is regular.
X		If L is the union of two decidable languages, then L is decidable.
	X	If L is the union of two undecidable languages, then L is undecidable. $A,\ \Sigma^* \setminus A$
	X	L is accepted by some NFA with 374 states if and only if L is accepted by some DFA with 374 states.
X		If $L \notin$ P, then L is not regular.

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

- Yes | No — $\{0^m 1^n \mid m \geq 0 \text{ and } n \geq 0\}$ 0^*1^*
- Yes | No — All strings with the same number of 0s and 1s
- Yes | No — Binary representations of all prime numbers less than 10^{100}
- Yes | No — $\{ww \mid w \text{ is a palindrome}\}$
- Yes | No — $\{wxw \mid w \text{ is a palindrome and } x \in \{0,1\}^*\}$ $w = \varepsilon$
- Yes | No — $\{\langle M \rangle \mid M \text{ accepts a finite number of non-palindromes}\}$

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

- Yes | No — $\varnothing$
- Yes | No — $\{0^n 1^{2n} 0^n 1^{2n} \mid n \geq 0\}$
- Yes | No — $\{ww \mid w \text{ is a palindrome}\}$
- Yes | No — $\{\langle M \rangle \mid M \text{ accepts a finite number of non-palindromes}\}$
- Yes | No — $\{\langle M \rangle \# w \mid M \text{ accepts } ww\}$
- Yes | No — $\{\langle M \rangle \# w \mid M \text{ accepts } ww \text{ in at most } |w|^2 \text{ steps}\}$

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

- Yes | No — $\{\langle M \rangle \mid M \text{ accepts an infinite number of strings}\}$
- Yes | No — $\{\langle M \rangle \mid M \text{ accepts either } \langle M \rangle \text{ or } \langle M \rangle^R\}$
- Yes | No — $\{\langle M \rangle \mid M \text{ does not accept exactly 374 palindromes}\}$
- Yes | No — $\{\langle M \rangle \# w \mid M \text{ accepts } ww \text{ in at most } |w|^2 \text{ steps}\}$

(e) Which of the following is a good English specification of a recursive function that can be used to compute the edit distance between two strings $A[1..n]$ and $B[1..n]$?

- Yes | No ✗ — *Edit*(i, j) is the answer for i and j.
- Yes | No ✗ — *Edit*(i, j) is the edit distance between $A[i]$ and $B[j]$.
- Yes | No ✗ — *Edit*$[1..n, 1..n]$ stores the edit distances for all prefixes.
- Yes ✗ | No — *Edit*(i, j) is the edit distance between $A[i..n]$ and $B[j..n]$.
- Yes | No ✗ — *Edit*$[i, j]$ is the value stored at row i and column j of the table.

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

$$\text{MUGGLE} := \{\langle M\rangle \mid M \text{ accepts } \texttt{SCIENCE} \text{ but rejects } \texttt{MAGIC}\}$$

Professor Potter, your instructor in Defense Against Models of Computation and Other Dark Arts, suggests a reduction from the standard halting language

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

Specifically, suppose there is a Turing machine DETECTOMUGGLETUM that decides MUGGLE. Professor Potter claims that the following algorithm decides HALT.

```
DECIDEHALT(⟨M, w⟩):
  Encode the following Turing machine:
      RUBBERDUCK(x):
        run M on input w
        if x = MAGIC
            return FALSE
        else
            return TRUE
  return DETECTOMUGGLETUM(⟨RUBBERDUCK⟩)
```

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

- Yes ✗ | No — If M rejects w, then RUBBERDUCK rejects MAGIC.
- Yes ✗ | No — If M ~~accepts~~ halts on w, then DETECTOMUGGLETUM accepts ⟨RUBBERDUCK⟩.
- Yes | No ✗ — If M rejects w, then DECIDEHALT rejects $\langle M\rangle\#w$.
- Yes ✗ | No — DECIDEHALT decides the language HALT. (That is, Professor Potter's reduction is actually correct.)
- Yes | No ✗ — DECIDEHALT actually runs (or simulates) RUBBERDUCK.

(g) Consider the following pair of languages:

- $\textsc{HamPath} := \left\{G \;\middle|\; G \text{ is a directed graph with a Hamiltonian path}\right\}$
- $\textsc{Acyclic} := \left\{G \;\middle|\; G \text{ is a directed acyclic graph}\right\}$

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

Yes	No	$\textsc{Acyclic} \in \text{NP}$
Yes	No	$\textsc{Acyclic} \cap \textsc{HamPath} \in \text{P}$
Yes	No	$\textsc{HamPath}$ is decidable.
Yes	No	There is no polynomial-time reduction from $\textsc{HamPath}$ to $\textsc{Acyclic}$.
Yes	No	There is no polynomial-time reduction from $\textsc{Acyclic}$ to $\textsc{HamPath}$.

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

Your cousin Elmo is visiting you for Christmas, and he's brought his favorite card game. The game is played with a row of n cards, each labeled with an integer (which could be positive, negative, or zero). Both players can see all n card values. On each turn, the current player must take the leftmost card, but then can choose to either keep the card or give it to their opponent. If they keep the card, their turn ends and their opponent takes the next card; however, if they give the card to their opponent, *the current player's turn continues* with the next card. In short, the player that does *not* get the ith card decides who gets the $(i+1)$th card. The game ends when all cards have been played. Each player adds up their card values, and whoever has the higher total wins.

For example, suppose the initial cards are $[3, -1, 4, 1, 5, -9]$, and Elmo plays first. Then the game might proceed as follows:

- Elmo keeps the 3, ending his turn.
- You give Elmo the −1.
- You keep the 4, ending your turn.
- Elmo gives you the 1.
- Elmo gives you the 5.
- Elmo gives you the −9. All cards are gone, so the game is over.
- Your score is $1+4+5-9=1$ and Elmo's score is $3-1=2$, so Elmo wins!

DP

Describe an algorithm to compute the highest possible score you can earn from a given row of cards, assuming Elmo plays first and plays perfectly. Your input is the array $C[1..n]$ of card values. For example, if the input is $[3, -1, 4, 1, 5, 9]$, your algorithm should return the integer 1.

Let MaxScore(i, me)
= max score I can get from C[i..n]
my turn if me=T
Elmo's if me=F

$$\text{MaxScore}(i, me) = \begin{cases} 0 & \text{if } i > n \\ \max\left\{ \begin{array}{l} C[i] + \text{MaxScore}(i+1, F) \\ \text{MaxScore}(i+1, T) \end{array} \right\} & \text{if } me = T \\ \min\left\{ \begin{array}{l} \text{MaxScore}(i+1, T) \\ C[i] + \text{MaxScore}(i+1, F) \end{array} \right\} & \text{if } me = F \end{cases}$$

See page 8

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

There are n galaxies connected by m intergalactic teleport-ways. Each teleport-way joins two galaxies and can be traversed in both directions. Also, each teleport-way uv has an associated cost of $c(uv)$ galactic credits, for some positive integer $c(uv)$. The same teleport-way can be used multiple times in either direction, but the same toll must be paid every time it is used.

Judy wants to travel from galaxy s to galaxy t, but teleportation is rather unpleasant, so she wants to minimize the number of times she has to teleport. However, she also wants the total cost to be a multiple of 10 galactic credits, because carrying small change is annoying.

Describe and analyze an algorithm to compute the minimum number of times Judy must teleport to travel from galaxy s to galaxy t so that the total cost of all teleports is an integer multiple of 10 galactic credits. Your input is a graph $G = (V, E)$ whose vertices are galaxies and whose edges are teleport-ways; every edge uv in G stores the corresponding cost $c(uv)$.

*[Hint: This is **not** the same Intergalactic Judy problem that you saw in lab.]*

Shortest path

Define $H = (V', E')$ directed

$V' = V \times \{0, 1, \ldots 9\}$

(v, r) means Judy is at v spent r mod 10 bux

$E' = \{(u, r) \to (v, r + c(uv) \bmod 10) \mid uv \in E,\ r \in \{0, 1 .. 9\}\}$

$c((u, r) \to (v, r')) = 1$

We need shortest path in H from $(s, 0)$ to $(t, 0)$

BFS

$O(V' + E') = O(V + E)$ time

$V' = 10V$

$E' = 10E$

CS/ECE 374 A ✧ Spring 2018	Gradescope name:
Fake Final Exam Problem 4	

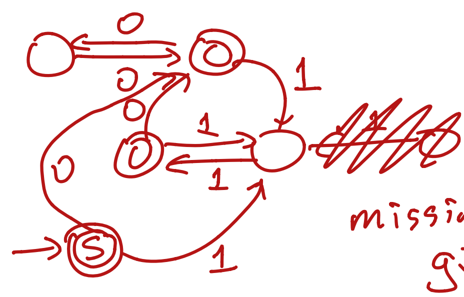
Recall that a ***run*** in a string is a maximal non-empty substring in which all symbols are equal. For example, the string 0001111000 consists of three runs: a run of three 0s, a run of four 1s, and another run of three 0s.

(a) Let L be the set of all strings in $\{0, 1\}^*$ in which every run of 0s has odd length and every run of 1s has even length. For example, L contains ε and 00000 and 0001111000, but L does not contain 1 or 0000 or 110111000.

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

odd run of 0s: $\to 0(00)^*$

even run of 1's: $\to 11(11)^*$



missing transitions go to junk state

$(\varepsilon + 11(11)^*)(0(00)^* 11(11)^*)^*(\varepsilon + 0(00)^*)$

(b) Let L' be the set of all non-empty strings in $\{0, 1\}^*$ in which the *number* of runs is equal to the *length* of the first run. For example, L' contains 0 and 1100 and 0000101, but L' does not contain 0000 or 110111000 or ε.

Prove that L' is not a regular language.

Fooling set

Let $F = (00)^+$

Choose x, y arbitrary in F

$x = 0^{2i}$ $y = 0^{2j}$ for some $i \neq j$ wlog $i < j$

Let $z = (10)^{i-1}1$

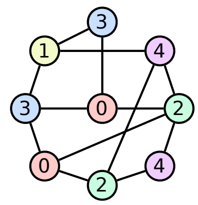
$xz = 0^{2i}(10)^{i-1}1$ has $2i$ runs so $xz \in L$

$yz = 0^{2j}(10)^{i-1}1$ has $2i$ runs so $yz \notin L$

so F is inf. fooling set for L ✓

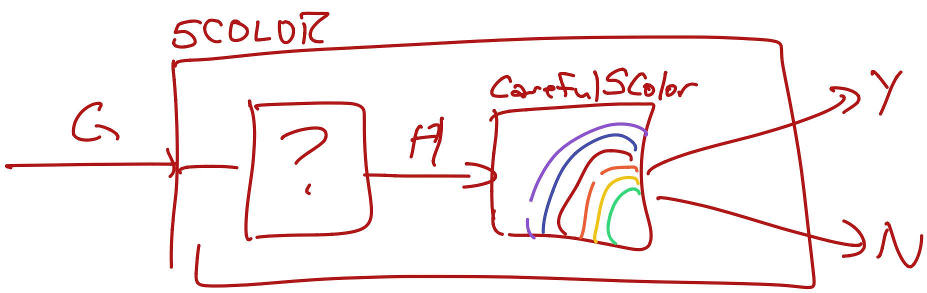
CS/ECE 374 A ✧ Spring 2018 Fake Final Exam Problem 5	Gradescope name:

Recall that a 5-coloring of a graph G is a function that assigns each vertex of G a "color" from the set $\{0,1,2,3,4\}$, such that for any edge uv, vertices u and v are assigned different "colors". A 5-coloring is *careful* if the colors assigned to adjacent vertices are not only distinct, but differ by more than 1 (mod 5).



A careful 5-coloring.

Prove that deciding whether a given graph has a careful 5-coloring is NP-hard. *[Hint: Reduce from the standard 5COLOR problem.]*



Let G be arbitrary graph
Define H as follows:
replace every edge uv in G with (u)—( )—( )—(v)
two new verts per edge

If G is 5-colorable
Fix a coloring
extend to careful of H:

wlog (1)—(2) ⟹ (1)—(4)—(0)—(2)
(1)—(3) ⟹ (1)—(3)—(1)—(3)

See page 7

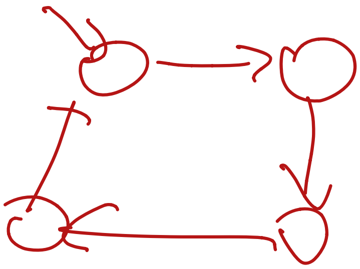
CS/ECE 374 A ✧ Spring 2018 **Fake Final Exam Problem 6**	Gradescope name:

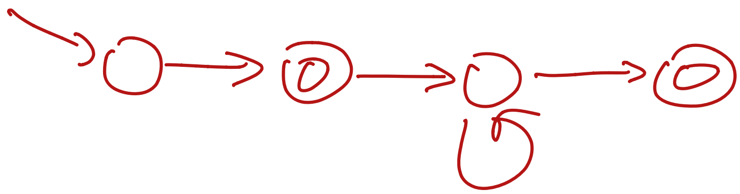
Describe and analyze an algorithm to determine whether the language accepted by a given DFA is finite or infinite. *[Hint: DFAs are directed graphs.]*

---

cycles? + reachability

Graphs



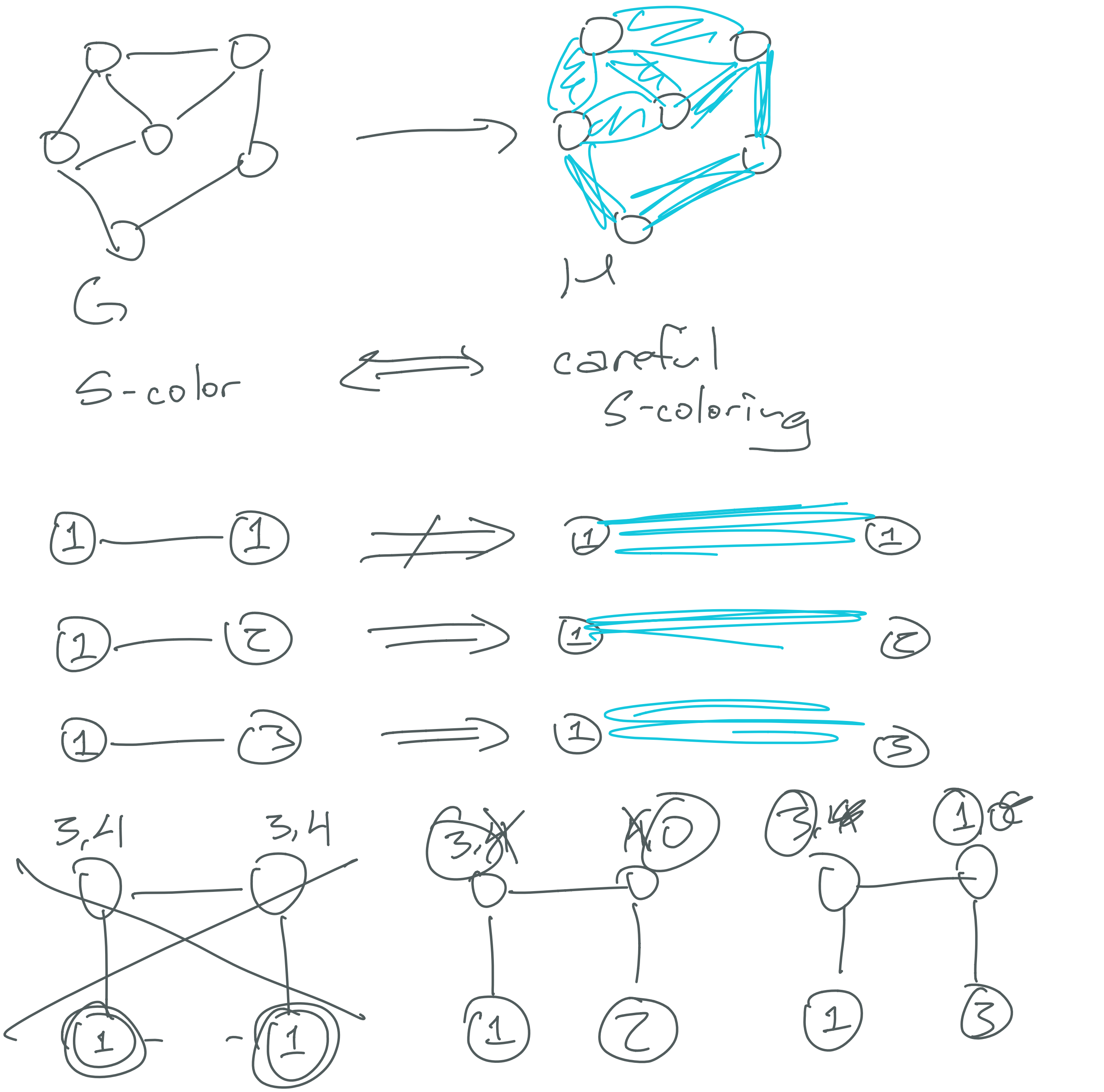


G has a cycle
reachable from s
can reach accepting state?

① kill verts not reachable from s
WFS(s) O(V+E) time ← total

② kill verts can't reach acc. state
add edge q→t new vertex t
for all q∈A
WFS(t) in rev(G) O(V+E)

③ Is remaining graph acyclic?
DFS/top sort O(V+E)



Spose H careful coloring
restriction to G is 5coloring because

can't be careful.

poly time ✓

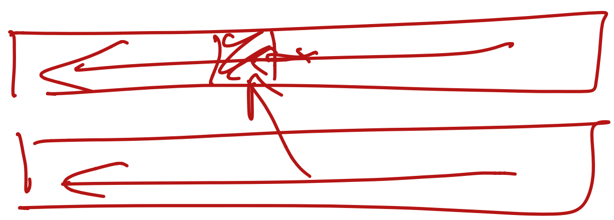
(scratch paper)

Elmo

Memoize into 2 1d arrays

MaxScoreMe [1..n+1] MS[i,T]

MaxScoreElmo [1..n+1] MS(i,F)



For i ← n down to 1

MSMe[i] ← . . .

MSEl(i) ← . . .

O(n)

We need ~~MaxScoreElmo[1]~~

MaxScore(1,T)

(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 $\max\{c \cdot x \mid Ax \le b, x \ge 0, x \in \mathbb{Z}^d\}$.

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 $\max\{x \in \mathbb{Z}^d \mid Ax \le b, x \ge 0\}$ 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?