CS500 Spring 2014
midterm

Hayes'
answer key

1. Let $A = \{0, 1\}$, and let $L \subseteq A^*$ consist of all numbers n, written in binary, such that n is a multiple of 7 or is congruent to 2 or 3 modulo 13. Prove that L is regular. What can be said about the size of the minimal DFA for L?

In class, saw how to make DFA to recognize numbers with given remainders mod m. Size was $O(m^2)$, (indeed $\le m(m-1)$).

Sln 1: Thus there is a DFA for (multiples of 7) and one for ($\equiv$ 2 or 3 (mod 13)).

Regular languages are closed under union, so L is regular.
That construction multiplies the sizes, so our DFA has $13 \cdot 12 \cdot 7 \cdot 6$ states (not quite minimal, but close).

Sln2: The constraints on remainders mod 7 and mod 13 amount to a constraint on remainders mod $7 \cdot 13 = 91$.
(Remainder must be 0, 7, 14, 21, ... or 2, 3, 15, 16, ... 80, 81).
So $\exists$ a DFA of size $\le 91 \cdot 90$ by the construction from class.

Myhill-Nerode Thm

Actually, 91 works. Should correct any confusion...

Need m states, since - when m is prime
when $a \not\equiv b$ mod m,
can pad w/ enough 0's to keep mod the same, then add in $(X-a)$ to bring total to X vs. $X + b - a$.

2. For a language L, let $f_L(n) = |\{i\colon 1 \le i \le n \text{ and } L \text{ contains at least one word of length } i\}|$. Say L *has sparse lengths* if

$$\lim_{n\to\infty} \frac{f_L(n)}{n} = 0.$$

Prove that no infinite context-free language has sparse lengths.

The Pumping Lemma for CFL's implies that any infinite CFL contains a "pumpable" word $w = uvxyz$, where $\forall t \ge 0$ $uv^t x y^t z \in L$ and $|vy| > 0$. If $|w| = a$ and $|vy| = b$, then all lengths of the form $a + ib$, where $i \ge -1$ are in the set defining f_L.

In particular, $f_L(a+ib) \ge i+2$.

But now $\lim\limits_{n\to\infty} \frac{f_L(n)}{n} \ge \lim\limits_{i\to\infty} \frac{f_L(a+ib)}{a+ib} \ge \lim\limits_{i\to\infty} \frac{i+2}{a+ib} = \frac{1}{b} \ne 0.$

So L doesn't have sparse lengths.

3. Let $L \subseteq \{0,1\}^*$.
(a) Define what it means for two words to be L-equivalent.
(b) Suppose all words are L-equivalent. What can be said about L?
For the remaining parts, suppose no two words are L-equivalent.
(c) Can L be regular?
(d) Can L be 1-DCA?
(e) Can L be context-free?

a) $u \sim_L v$ if $\forall w \in A^*, (uw \in L \Leftrightarrow vw \in L)$.

b) L is either $\emptyset$ or A^*. Why? Because if any words u, v satisfied $u \in L$ but $v \notin L$, then $u \not\sim_L v$ considering w = empty string as witness.

c) No. By Myhill-Nerode Thm, regular languages have only finitely many L-equivalence classes.

d) No, we know that the number of length-n L-equivalence classes grows as $O(n)$. But the number of length-n strings is 2^n.

e) Yes, PALINDROMES was an example of this, seen in class.

4. **Recognizing Non-Empty Regular Languages.** Consider the following problem:

Input: A DFA, $(S, A, s^0, S^{\text{yes}}, \delta)$.

Output: YES, if there exists at least one string accepted by the DFA.

How hard is this problem? Specifically, is it computable? In NP? In P?

Yes, this problem is in P. (In fact, even in LOGSPACE, which we haven't covered yet.) It amounts to asking whether there exists a path from s^0 to any vertex of S^{yes}. This can be checked using breadth-first search (or depth-first), in linear time.

(The labels on this path tell you a string that is accepted.)

5. The Triangle Cover problem is this:

Input: A graph $G = (V, E)$, and a positive integer k.

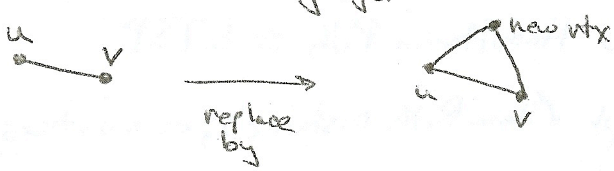
Output: YES, if there exists a set $S \subseteq V$ such that $|S| \leq k$ and every triangle in G includes at least one element of S. NO, otherwise.

Prove that Triangle Cover is NP-complete.

We know Vertex Cover is NPC, so it suffices to show

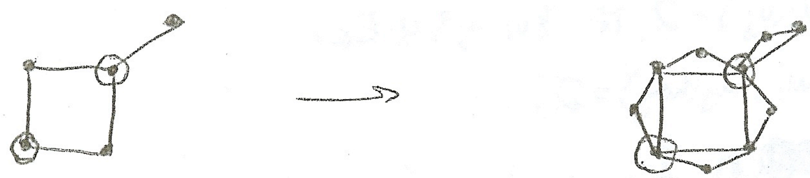
Vtx-Cover $\leq$ Triangle Cover.

The reduction uses this gadget:



So, given G with n verts, m edges, target size k $f(G)$ has $n+m$ verts, $3m$ edges. target size k.

Example



(vtx cover) (size 2 cover is circled in each) (triangle cover).

Check: G has vtx cover of size $k \Rightarrow f(G)$ has tri. cover of size k.

✓ because the same set S works.

$f(G)$ has tri. cover of size $k \Rightarrow G$ has vtx cover of size k.

Proof: Suppose S is a tri. cover of size k in $f(G)$. If S includes any new vtx, swap it for one of the endpoints of the corresponding edge. This is still a triangle cover, because a new vtx is only involved in 1 triangle. This yields a set $S' \subseteq V$ that is still size $\leq k$ and still a triangle cover. In particular, it covers all the edges of G, each of which is part of its own triangle containing only 2 vertices from G and a new vtx.

6. The Lazy Salesman. In the usual Travelling Salesman Problem, the salesman needs to visit all n of the given cities, ending where he starts. Suppose a lazy salesman only desires to visit any $n/2$ of the cities, and doesn't need to end where he starts. Is this variant of TSP in NP? Is it NP-hard?

LTSP $\in$ NP.

Proof: We can use the cheapest sequence of $n/2$ cities as our witness. Checking that the cost is within budget is obviously polytime.

LTSP is NPC.

Proof: I will show Hamiltonian Path $\leq$ LTSP.

Let G be any graph (Ham. Path instance), on n vertices, $v_1, v_2, \ldots, v_n$.

$f(G)$ adds n additional vertices $w_1, w_2, \ldots, w_n$.

Edge costs: $C(v_i, v_j) = 1$ if $\{v_i, v_j\} \in E_G$.

$C(v_i, v_j) = 2$ if $\{v_i, v_j\} \notin E_G$.

$C(w_i, \text{anything}) = 2$.

Target cost $= n-1$.

Claim: G has a Ham. path $\Leftrightarrow f(G) \in$ LTSP.

Proof is an easy exercise.