University of Nevada, Las Vegas Computer Science 456/656 Spring 2021

Answers to Assignment 4: Due Tuesday March 23, 2021

1. Which of these languages (problems) are **known** to be $\mathcal{NP}$-complete? If a language, or problem, is known to be $\mathcal{NP}$-complete, fill in the first circle. If it is either known not be be $\mathcal{NP}$-complete, or if whether it is $\mathcal{NP}$-complete is not known at this time, fill in the second circle.

   ● ○ Boolean satisfiability.
   ○ ● 2SAT.
   ● ○ 3SAT.
   ● ○ 4SAT.
   ● ○ Subset sum problem.
   ○ ● Generalized checkers, *i.e.* on a board of arbitrary size.
   ● ○ Independent set problem.
   ● ○ Traveling salesman problem.
   ○ ● Regular expression equivalence.
   ○ ● C++ program equivalence.
   ○ ● Rush Hour: https://www.youtube.com/watch?v=HI0rlp7tiZ0
   ○ ● Circuit value problem, CVP.
   ● ○ Regular grammar equivalence.
   ● ○ Dominating set problem.
   ● ○ Partition.

2. State the pumping lemma for regular languages.

   For any regular langauge L, there is an integer $p > 0$, which we call the pumping length of L, such that, for any string $w \in L$ of length at least p, there exist strings x, y, z such that the following four conditions hold.

   1. $w = xyz$,
   2. $|xy| \leq p$,
   3. y is not empty,
   4. For any integer $i \geq 0$, $xy^iz \in L$.

3. Use the pumping lemma to prove that the language L generated by the grammar given below is not regular.

   $S \to iS$
   $S \to iSeS$
   $S \to wS$
   $S \to a$

   By contradiction. Assume L is regular. Pick p a pumping length of L. Let $w = i^p a(ea)^p$, a member of L. Note tha $|w| \geq p$. By the pumping lemma, there exist strings x, y, z such that the following four conditions hold:

1. $w = xyz$,
2. $|xy| \leq p$,
3. y is not empty,
4. For any integer $i \geq 0$, $xy^iz \in L$.

Since $xyz = w$ and $|xy| \leq p$, xy is a substring of i^p, hence $y = i^k$ for some $k > 0$. By 4., $xy^0z = xz \in L$; $xz = i^{p-k}a(ae)^p$ which is not in L, since no member of L can have fewer i's than e's.

4. Let L be the language over $\{a, b\}$ consisting of all strings which have the same number of a's as b's, such as aabb, abba, aaabbb, bbbaaa, ... Design a PDA which accepts L.

   Since you can't easily draw figures during a lockdown browser exam, I will describe the states and transitions of PDA M in text.

   M has two states, the start state, and a second state which is final. We let z be the bottom up stack symbol. There are six transitions from the start state to itself. The labels of these transitions are: $z/a/az$, $z/b/bz$, $a/a/aa$, $a/b/\lambda$, $b/a/\lambda$, and $b/b/bb$. There is one transition from the start state to the second state, with label $z/\lambda/\lambda$.

5. Give a context-sensitive grammar for $\left\{a^{2^n}\right\} : n \geq 0$

   There are many correct answers. I think this one works:

   $S \to a|aa|aaaa|LAaR$
   $L \to LA|A$
   $Aa \to aaA$
   $AaR \to aaR$
   $LAa \to aaA$
   $AaaR \to aaaa$

6. Give a polynomial time reduction of the subset sum problem to partition.

   This is done in a posted document.

7. Give a polynomial time reduction of 3SAT to the independent set problem.

   This is done in a posted document.

8. Prove that a language is recursively enumerable, $\mathcal{RE}$, if and only if it is accepted by some machine.

   Suppose a language L is recursively enumerable by some machine M_1. Then L is accepted by some machine M_2

   Let $w_1, w_2, \ldots$ be the output of M_1, an enumeration of L. Let M_2 be a machine equivalent to the following program.

   Read w.
   For i from 1 to ∞
     If $w_i = w$
       Write "Accept"

   Then M_2 accepts L.

Conversely, suppose a language L over an alphabet Σ is accepted by a machine M_1. Let $w_1, w_2, \ldots$ be an enumeration of Σ^* in canonical order. Let M_2 be a machine equivalent to the following program.

```
For t from 1 to ∞
  For i from 1 to t
    If M_1 accepts w_i within t steps
      Write w_i
```

Since every $w \in L$ is equal to w_i for some i, and is accepted by M_1 in some number of steps, say t steps, w will be written by M_2 during the j^{th} iteration of the outer loop, where $j = \max(i, t)$.