Answers to Practice for the CS456 Final Examination: Part II

1. Prove that the language $L = \{a^n b^n : n \geq 0\}$ is not regular.

   Assume L is regular. Thus, L satisfies the conditions of the pumping lemma for regular languagues. Let p be the pumping length of L. Let $w = a^p b^p \in L$. Since $|w| = 2p$, there exist strings x, y, z such that: 1. $w = xyz$ 2. $|xy| \leq p$ 3. $|y| \geq 1$ 4. For any integer $i > 0$, $xy^i z \in L$. Since xy is a prefix of $a^p b^b$ of length at most p, $y = a^k$ for some $k > 0$. Let $i = 0$. Then $xy^i z = a^{p-k} b^p \notin L$, contradiction.

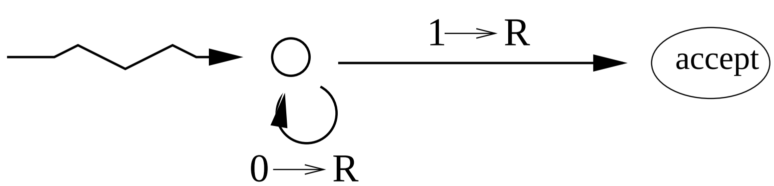
2. Give a context-sensitive grammar for $L = \{a^n b^n c^n : n \geq 1\}$ The grammar I gave on the previous version of this document was wrong. Here is my latest attempt:
   $S \to abc | aAbc$
   $aAb \to aabbB | aaAbbB$
   $Bb \to bB$
   $Bc \to cc$

3. Give a context-free language whose complement is not context-free.

   There are many examples. Here is one. $L = \left\{a^i b^j c^k : i \neq j \text{ or } j \neq k\right\}$ We have mentioned this example several times this semester.
   Let $L_1 = \left\{a^i b^j c^k : i < j\right\}$.
   Let $L_2 = \left\{a^i b^j c^k : i > j\right\}$.
   Let $L_3 = \left\{a^i b^j c^k : j < k\right\}$.
   Let $L_4 = \left\{a^i b^j c^k : j > k\right\}$.
   Then $L = S_1 + S_2 + S_3 + S_4$. Each of the L_i is clearly conext-free, so L is context-free. Now let L' be the complement of L, that is, $L' = \{a, b, c\}^* \backslash L$.

   Assume L' is context-free. Let R be the regular language described by the regular expression $a^* b^* c^*$. Then $L' \cap R$ is context-free, since it is the intersection of a context-free language with a regular language. But $L' \cap R = \{a^n b^n c^n : n \geq 0\}$, which, by the pumping lemma, is not context-free. Contradiction.

4. Write a regular expression for the binary language accepted by the Turing machine illustrated below.



   $0^* 1 (0+1)^*$

   That is, the machine accepts any binary string which is not all zeros.

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

   We define a reduction R of the subset sum problem to the partition problem. Let $w = (K, x_1, \ldots x_n)$ be an instance of the subset sum problem. Without loss of generality, $x_i > 0$ for all i. Let $S = \sum_{i=1}^{n} x_i$.

Without loss of generality, $K \leq S$. Let $R(w) = (x_1, \ldots x_n, K+1, S-K+1)$, an instance of the partition problem. Then w has a solution if and only if $R(w)$ has a solution. Clearly, R is polynomial time.

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

   We define a reduction R of 3SAT to the independent set problem. Let $w = C_1 * C_2 * \cdots C_m$, where $C_i = (t_{i,1} + t_{i,2} + t_{i,3})$ for all $1 \leq i \leq m$, and where $t_{i,j}$ either $I_{i,j}$ or $!I_{i,j}$ where $I_{i,j}$ is an identifier, an instance of 3SAT. We define $R(w) = (m, G)$, where G is the graph (V, E), where the set of vertices $V = \{v_{i,j} : 1 \leq i \leq m, 1 \leq j \leq 3\}$ and the set of edges E consist of the following pairs:
   (a) $(v_{i,1}, v_{i,2}), (v_{i,1}, v_{i,3}), (v_{i,2}, v_{i,3})$ for all i
   (b) $(v_{i,j}, v_{k,\ell}$ such that $t_{i,j} * t_{k,\ell}$ is a contradiction.

   Then w has a satisfying assignment if and only if G has an independent set of size m.

7. Prove that the language L given in problem 2 is not context-free.

   Assume L is context-free. Then L satisfies the conditions of the pumping lemma for context-free languages. Let p be the pumping length of L. Let $w = a^p b^p c^p \in L$. Since $|w| \geq p$, there exist strings u, v, x, y, z such that
   1. $w = uvxyz$
   2. $|vxy| \leq p$
   3. $|vy| \geq 1$
   4. For any integer $i \geq 0$, $uv^ixy^iz \in L$. Since vxy has length no longer than p, it cannot contain both an a and a c. Without loss of generality, it contains no a. Let $k = |vy| \geq 1$, and let $i = 0$. Then $uxz \in L$. But uxz must contain at most $p-1$ a's, and also must contain p c's. Thus $uxz \notin L$, contradiction.

8. Prove that the halting problem is undecidable.

   We are given system of encoding Turing machines as strings. The halting problem is the language $L_{\text{HALT}} = \{\langle M \rangle w : w \in L(M)\}$, where $\langle M \rangle$ is the encoding of the Turing machine M. Assume L_{HALT} is decidable. Let $L_{\text{DIAG}} = \{\langle M \rangle : \langle M \rangle \langle M \rangle \notin L_{\text{HALT}}\}$ Then L_{DIAG} is decidable because L_{HALT} is decidable. Thus, there is a Turing machine M_{DIAG} which accepts L_{DIAG}. If M is any Turing machine,

   1. For any Turing machine M, $\langle M \rangle \in L_{\text{DIAG}}$ if and only if $\langle M \rangle \langle M \rangle \notin L_{\text{HALT}}$, by definition of L_{DIAG}.

   2. For any Turing machine M, $\langle M \rangle \in L_{\text{DIAG}}$ if and only if $\langle M_{\text{DIAG}} \rangle \langle M \rangle \in L_{\text{DIAG}}$, by definition of M_{HALT}.

   By universal instantiation, we can replace $\langle M \rangle$ by $\langle M_{\text{DIAG}} \rangle$ in both of those statements. Thus

   1. $\langle M_{\text{DIAG}} \rangle \in L_{\text{DIAG}}$ if and only if $\langle M_{\text{DIAG}} \rangle \langle M_{\text{DIAG}} \rangle \notin L_{\text{HALT}}$.

   2. $\langle M_{\text{DIAG}} \rangle \in L_{\text{DIAG}}$ if and only if $\langle M_{\text{DIAG}} \rangle \langle M_{\text{DIAG}} \rangle \in L_{\text{HALT}}$.

   Contradiction.

9. The Dyck language, where left and right parentheses are replaced by a and b, is generated by the following unambiguous context-free grammar:

   1. $S \to S_{1,3} a_2 S_3 b_4$
   2. $S \to \lambda$

   Fill in the Action and Goto tables of an LALR parser for this grammar. I have done lines 0 and 4.

   The corrected table shows changes in column "a" and rows 2 and 3.

	a	b	$	S
0	r2		r2	1
1	s2		HALT	
2	r2	r2		3
3	s2	s4		
4	r1	r1	r1	

10. The following unambiguous context-free grammar generates the same language. Fill in the Action and Goto tables of an LALR parser for this grammar.

    1. $S \to a_2 S_3 b_4 S_5$
    2. $S \to \lambda$

	a	b	$	S
0	s2		r2	1
1			HALT	
2	s2	r2		3
3		s4		
4	s2	r2	r2	5
5		r1	r1	

11. The Dyck language contains the empty string. Let L be the Dyck language minus the empty string, which is generated by the CF grammar:

    1. $S \to aSb$
    2. $S \to SS$
    3. $S \to ab$

    Find a CNF (Chomsky Normal Form) grammar for L.

    Here is one possible answer.

    $S \to AT|AB|SS$
    $T \to SB$
    $A \to a$
    $B \to b$