CSE 311: Foundations of Computing I
Assignment #7
November 13, 2013
Due: November 20, 2013

Problems

1. Construct context-free grammars that generate the following sets of strings. For each of your constructions, write a sentence or two to explain why your construction is correct. If you use more than one variable, as documentation explain what sets of strings you expect each variable to generate.

   a. The set of all binary strings that contain at least two 0's and at most two 1's.

   b. The set of all binary strings that are of odd length and have 1 as their middle character.

2. If $a \in \Sigma$ is a symbol then the string a^n for $n \geq 0$ is the string consisting of n copies of a, one after the other. Construct context-free grammars that generate the following sets of strings. For each of your constructions, write a sentence or two to explain why it's correct. If you use more than one variable, as documentation explain what sets of strings you expect each variable to generate.

   a. $\{1^m 0^n 1^{m+n} : m, n \geq 0\}$

   b. $\{1^m 0^n 1^p : m, n, p \geq 0 \text{ and } m \geq n \text{ or } n \leq p\}$.

3. Define a grammar by $\boldsymbol{S} \to \boldsymbol{SS} \mid 0\boldsymbol{S}11 \mid 110\boldsymbol{S} \mid \lambda$. Use structural induction to prove that every string generated by $\boldsymbol{S}$ has twice as many 1's as 0's.

4. A relation R is called *circular* if $(c, a) \in R$ whenever $(a, b) \in R$ and $(b, c) \in R$. Prove that any reflexive relation R is circular if and only if it is both symmetric and transitive.

5. Let R be the relation on pairs of positive real numbers $\mathbb{R}^+ \times \mathbb{R}^+$ given by $\big((x, y), (u, v)\big) \in R$ if and only if $xv = uy$. Prove that R is reflexive, symmetric, and transitive.

6. A directed graph is called *acyclic* if it does not contain a directed cycle (a non-empty directed path from a vertex to itself). Show that for every directed acyclic graph G, the transitive-reflexive closure of the relation R represented by G is antisymmetric.

7. Is the transitive-reflexive closure of a relation R always equal to the transitive-reflexive closure of R^3? Justify your answer.

Extra credit:

Consider the following context-free grammar.

⟨Stmt⟩ → **⟨Assign⟩** | **⟨IfThen⟩** | **⟨IfThenElse⟩** | **⟨BeginEnd⟩**
⟨IfThen⟩ → if condition then **⟨Stmt⟩**
⟨IfThenElse⟩ → if condition then **⟨Stmt⟩** else **⟨Stmt⟩**
⟨BeginEnd⟩ → begin **⟨StmtList⟩** end
⟨StmtList⟩ → **⟨StmtList⟩** **⟨Stmt⟩** | **⟨Stmt⟩**
⟨Assign⟩ → `a := 1`

This is a natural-looking grammar for part of a programming language, but unfortunately the grammar is "ambiguous" in the sense that it can be parsed in different ways (that have different meanings).

a. Show an example of a string in the language that has two different parse trees.
b. Give a new grammar for the same language that is **unambiguous** in the sense that every string has a unique parse tree.