ECE 198 Fall 2019 Jin Yucheng (yucheng9)

Homework 5 Problem 1-2

Solution:

(1) Choose an undecidable problem that is not stated in terms of Turing Machines and provide a one-paragraph, less than half-page sketch of its undecidability proof.

Reference: "How is proving a context free language to be ambiguous undecidable?" *StackExchange*. 2011. <https://cstheory.stackexchange.com/questions/4352/how-is-proving-a-context-free-language-to-be-ambiguous-undecidable>

The undecidable problem I want to sketch the proof is, it is undecidable whether a context-free grammar is ambiguous.

The general idea is that we could reduce it from Post's Correspondence Problem (PCP). Suppose we can decide the language $L(G)$ generated by CFG G is ambiguous, we somehow construct a CFG and if the language is ambiguous then there is a derivation of some string in two different ways, but this leads to the result that we have a solution to PCP; similarly, if there is no ambiguity, then PCP cannot be solved. Therefore, we've reduced the original problem to PCP, and since PCP is undecidable, we can prove that it is undecidable whether a context-free grammar is ambiguous.

(2) Prove by induction that in a rooted tree, every node is reached by a DFS starting from the root.

A tree is a ***connected*** undirected graph with no simple circuits, and a rooted tree is a tree in which one vertex has been designated as the root and ***every edge is directed away from the root***. A preorder traversal is a DFS that visits all tree nodes, given a rooted tree T and its root r (this rooted tree is not empty), preorder traversal works as follows:

```
def PreorderTraversal($T, r$):
    visit $r$
    for each child $c$ of $r$ from left to right:
        $T_c$ = subtree with $c$ as its root
        PreorderTraversal($T_c, c$)
```

■

Proof of Correctness

- Basis Step: when the rooted tree T only has one node, r, then preorder traversal visits r.
- Inductive Step: Suppose preorder traversal visits all nodes of a rooted tree T', if T' is a subtree of another rooted tree T. Since T can be expressed as a root r with n children, each child i forms a subtree T_i. For any subtree, T_1, T_2, ..., T_n, preorder traversal visits all nodes, therefore, PreorderTraversal(T, r) first visits r and then calls PreorderTraversal(T_i, r_i) for every subtree i, and PreorderTraversal(T, r) visits r and every node in its subtrees.

So every node is reached by a DFS starting from the root in a rooted tree.