Problem Set 9

Handed out Wednesday, November 13, 2019

Due Wednesday, November 20, 2019, at 6 PM. Submit to LEARN.

In this assignment you don’t need to provide the complete details of any TM, but please provide enough information so that the marker can verify that your constructions work.

1. [10 marks] The goal of this exercise is to show that it doesn’t matter (for Turing decidability) if integers are represented in unary or in binary.

Let \( S \subseteq \mathbb{N} \) be a set of natural numbers. Define

\[
L_1 = \{a^n : n \in S\} \\
L_2 = \{\langle n \rangle : n \in S\}.
\]

Here \( a \) is a single symbol and, as usual, \( a^n = a \cdots a \), and \( \langle n \rangle \) denotes the binary representation of \( n \) (with no leading zeros), and \( \langle 0 \rangle = \epsilon \).

Show that \( L_1 \) is Turing-decidable iff \( L_2 \) is Turing-decidable.

2. [10 marks] Consider the language

\[
L = \{\langle M, n \rangle : M \text{ is a deterministic TM, } n \text{ is a positive integer, and there exists a string } x \text{ such that } M \text{ halts on input } x \text{ within } n \text{ moves}\}.
\]

Is \( L \) Turing-decidable? Prove or disprove.

3. [10 marks] Suppose we define a two-stack PDA to be like an ordinary PDA (allowed to be nondeterministic), except it has two independent stacks that it can manipulate. At every move, the PDA can choose to push or pop either one of its stacks, or read an input symbol, or do an \( \epsilon \)-move.

Argue that the class of languages recognized by two-stack PDA’s is the same as the class of languages recognized by Turing machines.

Your proof should show how to simulate a two-stack PDA with a TM, and vice versa.

Give enough details to show that you know what you’re doing, enough to convince the skeptical TA who will mark it.