Harvard University
Computer Science 121

Section 6 Notes
Week of 11.01.10

1 Outline

- Turing Machines
- Enumerators
- Equivalences
- Problems

2 Turing Machines

- We write a computation using configurations uqv (state q, head before v, tape holds uv).
- We use $\Rightarrow$ to denote one configuration leads to the next according to the transitions of the machine; $\Rightarrow^*$ for multiple steps.
- Many intuitive formulations are equivalent (multi-tape, nondeterministic, etc.).
- We prove by simulating any machine in one representation by some machine in the other.

3 Enumerators

- Recall, an enumerator prints the strings in a language.
- We define a language to be enumerable if it has an enumerator.
- A language is recursively enumerable iff it is enumerable.
- A language is recursive iff it is enumerable in lexicographic order.

4 Other equivalences

- Also equivalent to general grammar, 2 counters, etc.
- Church-Turing thesis tells us they are also equivalent to an "algorithm".
- But, when changing representation, running time or memory usage might change.

5 Problems

Exercise 5.1 *Write a general grammar that generates $\{a^i : i \text{ is composite}\}$.*

Exercise 5.2 *Write a general grammar that generates $\{a^{2^n} : n \geq 0\}$.*

Exercise 5.3 *Show that the language*

$$L = \{\langle M, w\rangle : M \text{ never moves its head left when running on } w\}$$

is decidable.

Exercise 5.4 *Let $L = \{\langle M\rangle : M$ is a TM that accepts at least one string in $\Sigma^*\}$. Show that L is recognizable.*

Exercise 5.5 *Let $L = \{\langle M\rangle : M$ is a TM that eventually writes a non-blank symbol when started on the empty tape$\}$. Show that L is decidable.*