6.045J/18.400J: Automata, Computability and Complexity Prof. Nancy Lynch, Nati Srebro

Recitation 5: Turing Machines

March 4, 2003 *Susan Hohenberger*

Problem 1: Define the following words, phrases and symbols.

1. Turing Machine
2. $(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$
3. (Turing) Decidable Language
4. (Turing) Recognizable Language
5. (Recursively) Enumerable Language
6. Halting

Problem 2: Mark each of the following statements either true or false.

1. A Turing machine has a single start state, but may have many accept states.
2. It is possible to make a Turing machine with only one state.
3. A Turing machine halts when its head reaches the end of its input.
4. All decidable languages are regular languages.
5. A nondeterministic TM with k-heads can recognize more languages than a deterministic TM with k-tapes.
6. A Turing machine might not halt on a finite input string.
7. A language L can be both co-decidable and undecidable.

Problem 3: (Sipser 3.7) It is stated in the text that Hilbert's 10th Problem (determining if a multivariable polynomial has an integer root) is undecidable. Here is a "pseudocode" description of an algorithm that takes in a polynomial, p, as input and determines if p has an integer root.

- For each possible setting of the variables $x_1, x_2, \ldots, x_k$ to integer values do:
  1. Evaluate p on $x_1, x_2, \ldots, x_k$.
  2. If p evaluates to 0, then accept, otherwise go to the next setting of $x_1, x_2, \ldots, x_k$.
- If none of the integer settings of $x_1, x_2, \ldots, x_k$ caused you to accept, then reject.

Why does this algorithm not contradict the fact that Hilbert's 10th Problem is undecidable?

Problem 4: Give a high-level description of a Turing machine T that decides, for a given representation $< M >$ of a DFA M, whether or not M accepts the word consisting of the single symbol 0. (This is what Sipser calls a "high-level description" on p. 145.)

(Bonus to think about.) If M is a Turing machine, rather than a DFA, would T still be a decider?

Problem 5: Describe the operation of a basic Turing machine that *recognizes* the language $L = \{ww^R : w \in \{0,1\}^*\}$. Explain how the Turing machine head(s) move and mark the tape(s), without listing the specific details of each transition. (This is what Sipser calls an "implementation description" on p. 145.)

How would you recognize the language $\{0^n 1 0^{2n} 1 0^{3n} : n \geq 1\}$ using a 3-headed, single tape TM?