6.045J/18.400J: Automata, Computability and Complexity Prof. Ron Rivest

Homework 4

Due: 13 March 2002

Problem 1: [**Turing Machines**] In this problem, we ask you to write out the specification of a Turing machine in the format used in Monday's lecture. (This is also the format used on pages 132 and 133.) Instead of recognizing a language, however, we want a Turing machine that will calculate a *function* of its input. That is, on input w, your Turing machine should always halt and accept, but when it does the tape should contain exactly $f(w)$ for some function f.

In this problem, the tape alphabet Γ will consist of three things:

- Σ, the input alphabet,
- $\dot{\Sigma}$, the characters of the input alphabet with dots over them, and
- Three special symbols: $\sqcup$ (space), x, and $\rightarrow$.

For this problem, the function f is defined to be the sequence of characters:

$$x^{|w|} \rightarrow W \rightarrow$$

where W is the word w with a space character $\sqcup$ inserted after each character of w and a dot over the first character of w. So, $f(01010)$ would be, for example:

$$xxxxx \rightarrow \dot{0} \sqcup 1 \sqcup 0 \sqcup 1 \sqcup 0 \sqcup \rightarrow$$

One last condition: your Turing machine can only write once to each cell of the tape. (It can overwrite the input, once per cell. Remember that at the beginning of computataion, the tape after the input is assumed to be filled with spaces.)

Be sure to explain the structure and operation of your machine.

After the above problem, you may explain the operations of your Turing machines at a much higher level of abstraction. See Examples 3.6, 3.7, 3.14, etc. Such descriptions will require additional explaination, of course, but are significantly easier to use and understand than the formal descriptions.

Problem 2: [**Turing Machine Variants**] Problem 3.10. (Hint: you can assume that there exists a Turing machine that can calculate the function of Problem 1, even if you couldn't specify it yourself.)

Problem 3: [**Enumerators**] Problem 3.16

Problem 4: [**Non-Determinism, Decidibility**] Problem 3.3