How this problem set will be graded:
You will be graded on the basis of whether your answers are correct.

Problems:

1. Design a Turing machine that does the following: when given a zero as input, it sets off on a task that never ends; when given a one as input, it halts.

2. Design a Turing machine that does the following: when given a string of $n$ ones as input, it produces a string of zeroes and ones that expresses $n$ in binary notation, and halts.

3. Design a Turing machine that does the following: when given a string of $n$ ones as input, it produces a string of one one if $n$ is prime and a string of two ones if $n$ is composite. (Assume $n \geq 2$)

4. Can any computation carried out by a Turing machine which is allowed to use two symbols (plus blanks) also be carried out by a Turing machine that is only allowed to use only one symbol (plus blanks)? If your answer is ‘no’, explain why not. If your answer is ‘yes’ explain how to transform a two-symbol Turing machine into the corresponding one-symbol Turing machine.

5. Extra Credit.

Design a Turing machine that does the following: when given a string of $n$ ones it produces a string of $m$ ones as output, where $m$ is the $n$th Ackermann Number. (See: http://en.wikipedia.org/wiki/Ackermann_function)

A Turing-machine simulator and instructions for building Turing-machines can be found at:

http://morphett.info/turing/turing.html

Although you should feel free to experiment with other simulators (which are widely available on the web), answers to problems 1–3 and 5 will only be given credit if the code is written in such a way that it can be copied and pasted into the simulator above.