Please indicate how much time you spent on this assignment.

This assignment is all about designing and working with Turing machines. Many of the problems require you *construct* a Turing machine to decide or recognize a language. The format required for these constructions is the simulator used in class:

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

(Remember that you need to select “semi-infinite tape” as the variant.)

The code for these Turing machines must be emailed as individual text file attachments to klingeti@grinnell.edu by 10:00am of the deadline. Please name the text files like this: `username-hw5-p1.tm` for “problem 1” where `username` is your Grinnell username. (You are also encouraged to comment your Turing machine code to give insight into how it works.)

**Problem 1**

Consider the language $A = \{010\}$ over the alphabet $\Sigma = \{0, 1\}$. (Yes this language contains only a single string.)

a) Construct a Turing machine that decides $A$.

b) Draw the diagram for $A$ as modeled in the textbook.

c) Does the previous diagram look like a DFA? Explain how you could show that all regular languages are decidable.

**Problem 2**

Let $\Sigma = \{a, b, c\}$ be the alphabet for this problem. Construct a Turing machine that decides the language

$$B = \{a^n b^n c^n \mid n \geq 0\}.$$
Problem 3

Let $\Sigma = \{0, 1\}$ be the input alphabet for this problem. Recall that the self-delimited pair of two bit strings $x$ and $y$ is the string

$$p(x, y) = 0^{|x|}1xy.$$

Construct a Turing machine that interprets its input as a self-delimited pair of strings $p(x, y)$ and then outputs the string $x$. By “output,” we mean that when the Turing machine halts, the only non-blank symbols left on the tape forms the string $x$. (Note that blank spaces to the left of $x$ are permitted.)

Problem 4

Construct a Turing machine that interprets its input as a self-delimited pair of strings $p(x, y)$ and then outputs the string $y$.

Bonus Problem:

Construct a Turing machine that interprets its input as a self-delimited pair of strings $p(x, y)$ and then outputs the maximum of $x$ and $y$. (The values being compared are the unsigned integer values of the bits of $x$ and $y$.)