CS 81 Assignment 8 for Wed., April 6

1. Work out the details of simulating an arbitrary Turing machine in logic, specifically Otter. The representation should be pure logic, along the lines of the pegs puzzle presented in the lecture. Do not use evaluable functions. Each transition rule should be encoded as a clause. When a designated one of your predicates has an argument term that is an encoding of the input tape, your machine should simulate the machine on that tape. The system should be able to produce a clause containing only the answer literal *iff* the TM will halt on the corresponding input. When the machine halts, the answer literal should contain an argument term representing the contents of the tape. In this problem, we are not going to differentiate between accepting and rejecting states; the answer is always whatever is on the tape when the machine halts.

   Demonstrate your result by constructing a Turing machine that will accept the language, where acceptance is defined by having a 1 somewhere on the tape.

   $\{a^n\ b^n\ c^n \mid n \in N\}$

   $= \{\Lambda$, abc, aabbcc, aaabbbccc, …$\}$

2. Based on your result in problem 1, discuss what, if anything, can be concluded about the set of valid formulas in predicate calculus?

   Classify the following problems as to whether they are computable or not, and give justification, ideally a proof, for your answers.

3. $$f(x) = \begin{cases} 1 & \text{if x is the description of a Turing machine having more than 10 control states} \\ 0 & \text{otherwise} \end{cases}$$

4. $$g(x) = \begin{cases} 1 & \text{if x is the description of a Turing machine that visits at most 10 cells when started on a blank tape} \\ 0 & \text{otherwise} \end{cases}$$

5. $$h(x) = \begin{cases} 1 & \text{if x is the description of a Turing machine that eventually visits control state } q_{10} \text{ when started on a blank tape} \\ 0 & \text{otherwise} \end{cases}$$

6. $$k(x) = \begin{cases} 1 & \text{if x is the description of a Turing machine that eventually halts if started on an input tape containing 1111111111} \\ 0 & \text{otherwise} \end{cases}$$