Automata Theory and Computability

Assignment 4 (PDA's and Decidability)

Due on Tue 1st December 2015.

1. This question is about the prefix-closure of the language accepted by a PDA. A *pushdown system* (PDS) is a PDA in which all states are final. Acceptance is assumed to be by final state.
   (a) Show that for a given PDA P we can construct a PDS P' that accepts the prefix closure of the language accepted by P.
   (b) Give a construction that preserves determinacy: i.e. if P was a DPDA then P' would be a DPDS.
2. Give a total Turing machine which accepts the language
   $$L = \{0^n 1 0^m \mid n \geq m\}.$$
   Give your machine in the form of a state diagram, as done in class.
3. Prove that the following question is undecidable: Given a Turing machine M and a state q of M, does M ever enter state q on *some* input?
4. Let REG be the language of (encodings of) Turing machines, which accept a regular language. Thus
   $$\text{REG} = \{M \mid L(M) \text{ is regular }\}.$$
   Show that neither REG nor its complement is recursively enumerable.
5. Is is decidable to check whether the language of a given PDA P is contained in that of a given VPA V? Justify your answer.
6. Is it decidable whether the complement of a given CFL is also a CFL? Justify your answer
7. In a couple of undecidability proofs based on *valcomps* done in class we used the following argument. To check whether two strings c_1 and c_2 over the alphabet $\Gamma \times (Q \cup \{-\})$ represent configurations of the TM M with c_2 a valid consecution of c_1, it was sufficient to check that the triple of symbols starting at each position i (for $i \in \{1, \ldots, (|c_1| - 2)\}$) in c_1 "matches" the triple at position i in c_2. You may assume that c_1 and c_2 have the same length.
   (a) Demonstrate that it is *not* sufficient to consider just *a pair* of symbols at each position.
   (b) Under what assumptions on c_1 and c_2 *is* it sufficient to consider pairs of symbols at each position?