1. (15 pts) Recall that counter machines are special cases of multistack machines in which each stack has only one type of symbol. Informally describe counter machines that accept the following languages. Your machines should be deterministic and you should not use more than two counters.
   
   (a) \{0^n1^m | n \geq m \geq 1\}
   
   (b) \{a^i b^j c^k | i = j or i = k\}
   
   (c) \{a^i b^j c^k | i = j or i = k or j = k\}

2. (15 pts) (9.1.3) The following two languages are very similar to \(L_d\), but are not equivalent. For each, use a "diagonalization-style" argument to show that the language is not accepted by a Turing machine. Note that your argument can't be based on the actual diagonal of the matrix of entries, but instead will utilize another infinite sequence of points in the matrix.
   
   (a) The set of all strings \(w_i\) such that \(w_i\) is not accepted by \(M_{2i}\).
   
   (b) The set of all strings \(w_i\) such that \(w_{2i}\) is not accepted by \(M_i\).

3. (10 pts) (9.1.4) We have only considered Turing Machines with input alphabets \(\{0, 1\}\). Suppose instead we wanted to assign an integer to all Turing machines, regardless of their input alphabet. This is not quite possible in the encoding that we used because the names of the input symbols matter—\(\{a^n b^n | n \geq 1\}\) is not the same language as \(\{0^n 1^n\}\). However, suppose that we have an infinite set of symbols, \(\{a_1, a_2, \ldots\}\) from which all TM inputs are chosen. Show how to assign an integer to each TM which uses a finite subset of these symbols as its input alphabet.

4. (10 pts) (9.2.1) Show that the Halting Problem (the set of all \((M, w)\) pairs such that when \(M\) is given the input string \(w\), \(M\) halts, with or without accepting) is RE but not recursive.

5. (10 pts) (9.2.2) Define a recursive function as a function \(F\) defined by a finite set of rules, where each rule specifies the value of the function \(F\) for certain arguments. For example, Ackermann’s function is defined by the rules: (Note that this rule set is different than other common versions, e.g. the one on Wikipedia...)
   
   (a) \(A(0, y) = 1\) for any \(y \geq 0\)
   
   (b) \(A(1, 0) = 2\)
   
   (c) \(A(x, 0) = x + 2\) for any \(x \geq 2\)
   
   (d) \(A(x + 1, y + 1) = A(A(x, y + 1), y)\) for any \(x \geq 0 y \geq 0\).
   
   (a) Evaluate \(A(2, 1)\).
   
   (b) What function of \(x\) is \(A(x, 2)\)?
   
   (c) Evaluate \(A(4, 3)\).