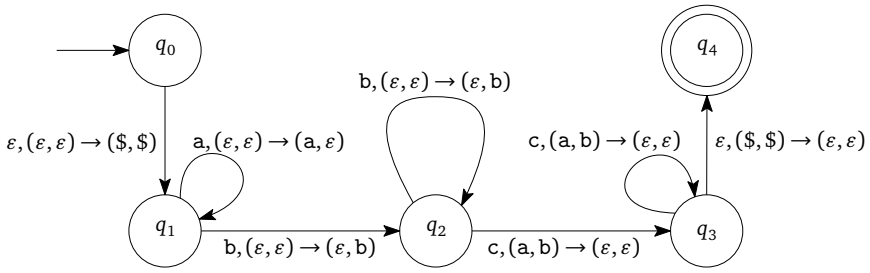
Prework 3.3a: The Church-Turing Thesis

Write your preliminary solutions to each problem and submit a PDF on Canvas. The names in brackets indicate the subset responsible for presenting the problem.

1. [Allie, David] Define an encoding of the polynomial $Ax^2 + Bx + C$ (with coefficients $A, B, C \in \mathbb{Z}$) as a string of the form $(0 \cup 1)^{|A|}\#(0 \cup 1)^{|B|}\#(0 \cup 1)^{|C|}$, where the 0's and 1's are chosen according to whether the corresponding coefficients are positive or negative. For example, $3x^2 - 4x + 2$ would encode as `000#1111#00`. What polynomial is encoded as `0#1#111111`? How would you encode the polynomial $x^2 - 4$?

2. [Grace, Connor] Give an implementation description of a TM (possibly multitape or nondeterministic) that decides the language of all strings of the form $0^n\#q$, where q is an encoding of the quadratic polynomial $Ax^2 + Bx + C$ (as defined in Problem 1), and n is a root of this polynomial (i.e., $An^2 + Bn + C = 0$).

3. [Micah, Ky] A *two-stack pushdown automaton* is defined similarly to a regular PDA, except that two stacks are maintained and can be pushed/popped at each transition. In the state diagram, arrows are labeled as $x, (s_1, s_2) \to (t_1, t_2)$ to indicate that x is the input symbol being consumed, s_1 and s_2 are the stack symbols being popped from stack #1 and stack #2, respectively, and t_1 and t_2 are the respective symbols to be pushed. Determine the language recognized by the following two-stack PDA.



4. [Meghan, Ben] Is the language recognized by the two-stack PDA in Problem 3 a context-free language? How do you know? What does this say about the computational power of two-stack PDA's?

5. [Levi, Todd, Joshua] Explain how to simulate a TM with a two-stack PDA. (For each type of transition of the TM (left or right), explain how the two stacks are managed in a way that is equivalent to writing on the TM's tape.)

6. [Andrew, Curtis] Explain how to simulate a two-stack PDA with a TM.

BEGIN YOUR SOLUTIONS BELOW THIS LINE