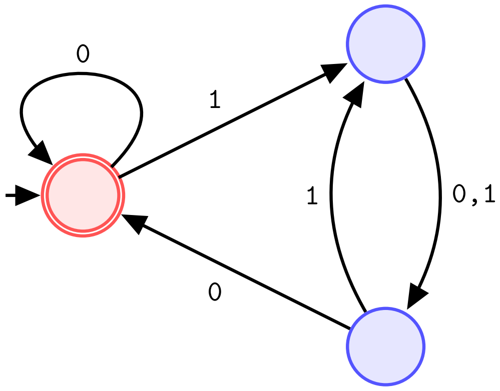
(1) Answer all parts for the following DFA M and give reasons for your answers.



1) Is $\langle M, \texttt{0100}\rangle \in A_{\text{DFA}}$?

2) Is $\langle M, \texttt{011}\rangle \in A_{\text{DFA}}$?

3) Is $\langle M\rangle \in A_{\text{DFA}}$?

4) Is $\langle M\rangle \in E_{\text{DFA}}$?

5) Is $\langle M, M\rangle \in EQ_{\text{DFA}}$?

(2) Draw a Venn diagram to illustrate the relationship between the languages: *regular, context-free, decidable*, and *Turing-recognizable*; then indicate where the following problems belong to. (Refer to the lecture slides.)

1. Acceptance problems
   i. $A_{\text{DFA}} = \{\langle B, w\rangle \mid B \text{ is a DFA that accepts input string } w\}$
   ii. $A_{\text{NFA}} = \{\langle B, w\rangle \mid B \text{ is an NFA that accepts input string } w\}$
   iii. $A_{\text{CFG}} = \{\langle G, w\rangle \mid G \text{ is a CFG that generates string } w\}$
   iv. $A_{\text{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$
2. Language emptiness problems
   i. $E_{\text{DFA}} = \{\langle A\rangle \mid A \text{ is a DFA and } L(A) = \emptyset\}$
   ii. $E_{\text{CFG}} = \{\langle G\rangle \mid G \text{ is a CFG and } L(G) = \emptyset\}$
3. Language equality problems
   i. $EQ_{\text{DFA}} = \{\langle A, B\rangle \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B)\}$
   ii. $EQ_{\text{CFG}} = \{\langle G, H\rangle \mid G \text{ and } H \text{ are CFGs and } L(G) = L(H)\}$

(3) Read about the *Post Correspondence Problem (PCP)* on Wikipedia (http://en.wikipedia.org/wiki/Post_correspondence_problem).

Is it decidable? How about when the alphabet is simply $\{\texttt{a}\}$?

Find some (easy) examples and try to solve them by hand, e.g.

$$\left\{ \left[\frac{\texttt{ab}}{\texttt{abab}}\right], \left[\frac{\texttt{b}}{\texttt{a}}\right], \left[\frac{\texttt{aba}}{\texttt{b}}\right], \left[\frac{\texttt{aa}}{\texttt{a}}\right] \right\}$$

Try to write code to search for solutions using brute force search. You may want to first have a look at: http://code.google.com/p/post-correspondence-brute/

(4) Let $AMBIGCFG = \{\langle G\rangle \mid G \text{ is an ambiguous CFG}\}$.

Show that $AMBIGCFG$ is undecidable.

Hint: Use a reduction form PCP (above). Given an instance

$$\left\{ \left[\frac{t_1}{b_1}\right], \left[\frac{t_2}{b_2}\right], \dots, \left[\frac{t_k}{b_k}\right] \right\}$$

of the Post Correspondence Problem, construct a CFG G with the rules:

$$\begin{array}{rcl} S & \to & T \mid B \\ T & \to & t_1 T a_1 \mid \cdots \mid t_k T a_k \mid E \\ B & \to & b_1 B a_1 \mid \cdots \mid b_k B a_k \mid E \\ E & \to & \varepsilon \end{array}$$

where $a_1, a_2, \dots, a_k$ are new terminal symbols.

"A *quine* is a computer program which takes no input and produces a copy of its own source code as its only output. The standard terms for these programs in the computability theory and computer science literature are *self-replicating programs*, *self-reproducing programs*, and *self-copying programs*." http://en.wikipedia.org/wiki/Quine_(computing)

Write a quine in your preferred programming language.