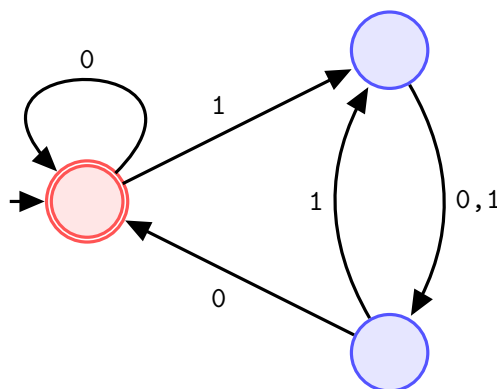


- (1) Answer all parts for the following DFA  $M$  and give reasons for your answers.



- Is  $\langle M, 0100 \rangle \in A_{\text{DFA}}$ ?
  - Is  $\langle M, 011 \rangle \in A_{\text{DFA}}$ ?
  - Is  $\langle M \rangle \in A_{\text{DFA}}$ ?
  - Is  $\langle M \rangle \in E_{\text{DFA}}$ ?
  - 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

- $A_{\text{DFA}} = \{ \langle B, w \rangle \mid B \text{ is a DFA that accepts input string } w \}$
- $A_{\text{NFA}} = \{ \langle B, w \rangle \mid B \text{ is an NFA that accepts input string } w \}$
- $A_{\text{CFG}} = \{ \langle G, w \rangle \mid G \text{ is a CFG that generates string } w \}$
- $A_{\text{TM}} = \{ \langle M, w \rangle \mid M \text{ is a TM and } M \text{ accepts } w \}$

2. Language emptiness problems

- $E_{\text{DFA}} = \{ \langle A \rangle \mid A \text{ is a DFA and } L(A) = \emptyset \}$
- $E_{\text{CFG}} = \{ \langle G \rangle \mid G \text{ is a CFG and } L(G) = \emptyset \}$

3. Language equality problems

- $EQ_{\text{DFA}} = \{ \langle A, B \rangle \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B) \}$
- $EQ_{\text{CFG}} = \{ \langle G, H \rangle \mid G \text{ and } H \text{ are CFGs and } L(G) = L(H) \}$

- (3) Read about the *Post Correspondence Problem (PCP)* on Wikipedia ([http://en.wikipedia.org/wiki/Post\\_correspondence\\_problem](http://en.wikipedia.org/wiki/Post_correspondence_problem)).

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

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

$$\left\{ \left[ \begin{array}{c} ab \\ abab \end{array} \right], \left[ \begin{array}{c} b \\ a \end{array} \right], \left[ \begin{array}{c} aba \\ b \end{array} \right], \left[ \begin{array}{c} aa \\ a \end{array} \right] \right\}$$

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 from PCP (above). Given an instance

$$\left\{ \begin{bmatrix} t_1 \\ b_1 \end{bmatrix}, \begin{bmatrix} t_2 \\ b_2 \end{bmatrix}, \dots, \begin{bmatrix} t_k \\ b_k \end{bmatrix} \right\}$$

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

$$\begin{aligned} S &\rightarrow T \mid B \\ T &\rightarrow t_1 T a_1 \mid \dots \mid t_k T a_k \mid E \\ B &\rightarrow b_1 B a_1 \mid \dots \mid b_k B a_k \mid E \\ E &\rightarrow \varepsilon \end{aligned}$$

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\\_\(computing\)](http://en.wikipedia.org/wiki/Quine_(computing))

Write a quine in your preferred programming language.