Stefanie Shidoosh

CS 181 HW#8

1. Let $\Sigma = \{ (\,,\,) \}$. $G = \{V, \Sigma, R, S\}$, $V = \{S, S_1\}$,
R: $\{ S \rightarrow SS_1 \mid S_1 \mid \varepsilon \quad , S_1 \rightarrow (S) \mid (\,)\}$

a. G is ambiguous. There exists more than one corresponding leftmost derivation for the same string, namely string w = ( ) ( ) ( ):

$S \Rightarrow S\,S_1 \Rightarrow SS_1S_1 \Rightarrow S_1S_1S_1 \Rightarrow (\,)\,S_1S_1\ldots$

OR

$S \Rightarrow S\,S_1 \Rightarrow SS_1S_1 \Rightarrow S_1S_1S_1 \Rightarrow (S)S_1S_1 \Rightarrow (\varepsilon)\,S_1S_1 \Rightarrow (\,)\,S_1S_1\ldots$

b. Show a leftmost reduction in G of the string w

$\underline{(\,)}\,(\,)\,(\,)$

$\underline{S_1}\,(\,)\,(\,)$

$S\,\underline{(\,)}\,(\,)$

$\underline{SS_1}\,(\,)$

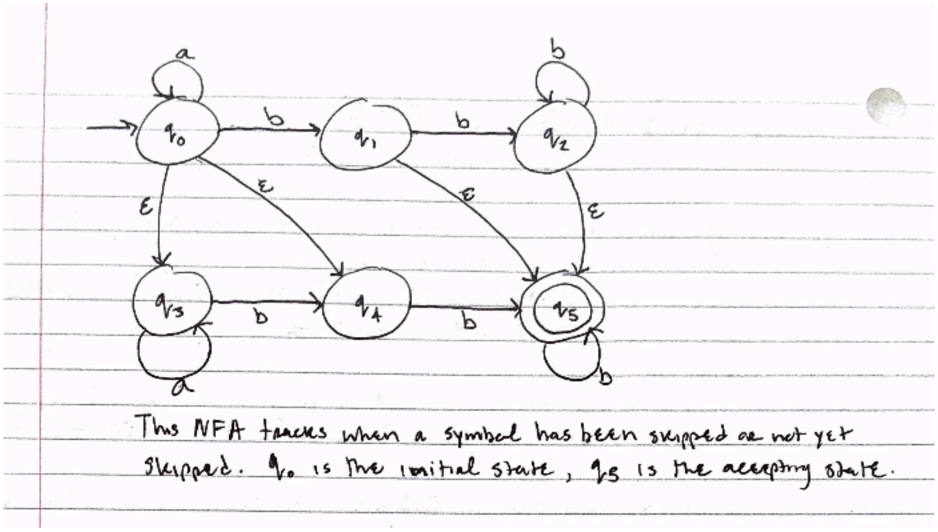
$S\underline{(\,)}$

$\underline{SS_1}$

S

Stefanie Shidoosh

2. NFA for DROPOUT(L) = {y in {a,b}* | xyz in L, for some x, z in {a, b}*}



Stefanie Shidoosh

3. Let $\Sigma = \{0, 1\}$, $P = \{w \mid w \text{ is a prime number in binary}\}$
Fact: We can devise a Turing Machine, MPE, that generates the prime numbers in the above form and writes each one on its work tape in strictly increasing numerical order from left to right separated by blanks. Thus, P is a member of the family of Recursively Enumerable (RE) languages.
Given all this information, what can we say about whether the language, P, is Recursive vs. RE but not Recursive? *Briefly* justify your answer, but do not spend time on a detailed proof.

A language is recursive if there exists a Turing machine that accepts every string of the language and rejects every string (over the same alphabet) that is not in the language. A language is recursively enumerable if there exists a Turing machine that accepts every string of the language, and does not accept strings that are not in the language. (Strings that are not in the language may be rejected or may cause the Turing machine to go into an infinite loop). In the case of P, every recursive language is recursively enumerable trivially, but the reverse is not so simple. In fact, the set of recursive languages is a proper subset of recursively enumerable languages (insert detailed proof we are not to spend time on here), hence P is recursively enumerable but not recursive.