

CS314 Fall 2018 Assignment1 Solution

September 2018

1 Rewrite System

a) $01 \implies \varepsilon$
 $10 \implies \varepsilon$

b) No, in some scenarios, it is possible that multiple rules can be applied.
E.g. using the two rules in a) and the the input sequence $\$1001\#$, if we use $01 \implies \varepsilon$, we can get $\$10\#$. But if we use $10 \implies \varepsilon$, we get $\$01\#$.

c)

$\$0101\#$	use rule 01 $\implies \varepsilon$
$\$01\#$	use rule 01 $\implies \varepsilon$
$\$ \#$	no rules
<hr/>	
$\$10110\#$	use rule 10 $\implies \varepsilon$
$\$110\#$	use rule 10 $\implies \varepsilon$
$\$1\#$	no rules

2 Regular Expressions

$C_const \implies int_const|fp_const$

$int_const \implies (oct_int|dec_int|hex_int)int_suffix$

$fp_const \implies (dec_fp|hex_fp)fp_suffix$

$oct_int \implies 0\ nonzero_oct_digit\ oct_digit^*$

$dec_int \implies nonzero_dec_digit\ dec_digit^*$

$hex_int \implies (0x|0X)\ nonzero_hex_digit\ hex_digit^*$

$dec_fp \implies sign\ dec_digit\ dec_digit^*\ dec_exponent| sign\ dec_digit^*$
 $(.dec_digit|dec_digit.)\ dec_digit^*\ (dec_exponent|\epsilon)$

$hex_fp \implies sign\ (0x|0X)\ hex_digit^*(.hex_digit|\epsilon|hex_digit.)hex_digit^*$
 $hex_exponent$

$int_suffix \implies \epsilon| unsigned_suffix(long_suffix|longlong_suffix|\epsilon)|$
 $long_suffix(unsigned_suffix|\epsilon)| longlong_suffix(unsigned_suffix|\epsilon)$

$fp_suffix \implies float_suffix|long_suffix|\epsilon$

$nonzero_oct_digit \implies 1|2|3|4|5|6|7$

$nonzero_dec_digit \implies nonzero_oct_digit|8|9$

$nonzero_hex_digit \implies nonzero_dec_digit|a|b|c|d|e|f|A|B|C|D|E|F$

$oct_digit \implies 0|nonzero_oct_digit$

$dec_digit \implies 0|nonzero_dec_digit$

$hex_digit \implies 0|nonzero_hex_digit$

$sign \implies \epsilon|+|-$

$dec_exponent \implies (e|E)sign\ dec_digit\ dec_digit^*$

$hex_exponent \implies (p|P)sign\ dec_digit\ dec_digit^*$

$unsigned_suffix \implies u|U$

$long_suffix \implies l|L$

$longlong_suffix \implies ll|LL$

$float_suffix \implies f|F$

3 Regular Expressions

- a) All binary strings, including the empty string.
- b) All binary strings of length ≥ 3 , with 1 as the third to the last digit.
- c) All binary strings of even number of 0's and 1's

4 Regular Expressions

- a) $(a|c)^*(b|c)^*$

Please be careful about the difference between "no a's following b's" and "no a's immediately following b's".

- b) $c^*(a|\varepsilon)c^*(a|\varepsilon)c^*(b|\varepsilon)c^*(b|\varepsilon)c^*|$
 $c^*(a|\varepsilon)c^*(b|\varepsilon)c^*(a|\varepsilon)c^*(b|\varepsilon)c^*|$
 $c^*(a|\varepsilon)c^*(b|\varepsilon)c^*(b|\varepsilon)c^*(a|\varepsilon)c^*|$
 $c^*(b|\varepsilon)c^*(a|\varepsilon)c^*(a|\varepsilon)c^*(b|\varepsilon)c^*|$
 $c^*(b|\varepsilon)c^*(a|\varepsilon)c^*(b|\varepsilon)c^*(a|\varepsilon)c^*|$
 $c^*(b|\varepsilon)c^*(b|\varepsilon)c^*(a|\varepsilon)c^*(a|\varepsilon)c^*$