CSI4101 Compiler Design
2nd Semester, 2019

Solutions to Homework 1

Note: no solution is given for Exercise 7. Exercise 7 is therefore the only exercise that you should hand in by the due-date.

1. Given the alphabet $\Sigma = \{-, +, x, y, z, 0, 1\}$ and the below regular expression, which strings are valid? (Mark one or more.)

$$R : (-| + |\epsilon)\ (x|y|(0|1)^*)z(0|1)^*$$

   (a) $-xz1$
   (b) $-xz$
   (c) yxz
   (d) ϵ
   (e) $01z - 1$
   (f) $+011z1$
   (g) $+x$
   (h) zx

   Answer: strings (a), (b) and (f) are valid.

2. Given the regular expression below, which strings are valid? (Mark one or more.) Note: "$[a - z]$" denotes lower-case characters, "$[A - Z]$" denotes upper-case characters, and "$[0 - 9]$" denotes the digits from 0–9.

$$R : [a - z]([a - z]|[A - Z]|[0 - 9]|\_)^*$$

   (a) $x\_$
   (b) $\_x$
   (c) $yx0z$
   (d) ϵ
   (e) $ada1$
   (f) $a\_d\_a\_1$

   Answer: strings (a), (c), (e) and (f) are valid.

3. Write regular expressions for the following languages whose alphabet is $\Sigma = \{0, 1\}$.

   (a) All possible strings, including the empty string.
   Answer: $(0|1)^*$

   (b) The empty string.
   Answer: ϵ

(c) The string 1011.
Answer: 1011

(d) The strings 1 and 101.
Answer: 1 | 101

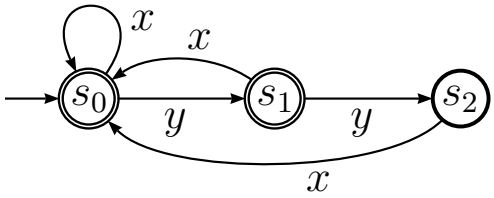
(e) All strings beginning with 01.
Answer: 01 (0|1)*

(f) All strings that contain exactly two 1's.
Answer: 0* 1 0* 1 0*

(g) All strings beginning with a 0 and ending with a 1.
Answer: 0 (0|1)* 1

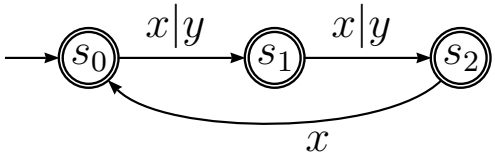
4. Design deterministic finite automata (DFAs) to recognize the following languages over the alphabet $\Sigma = \{x, y\}$.
Note that with the following DFAs we use an arrow to mark the starting state, e.g.,

(a) Every occurrence of the substring yy is followed by an x.
Answer:



- State s_0: every string where the last recognized symbol was not y.
- State s_1: every string where the last recognized symbol was y, but the one before was not y.
- State s_2: every string where the previous two recognized symbols were y, but the one before was not y.

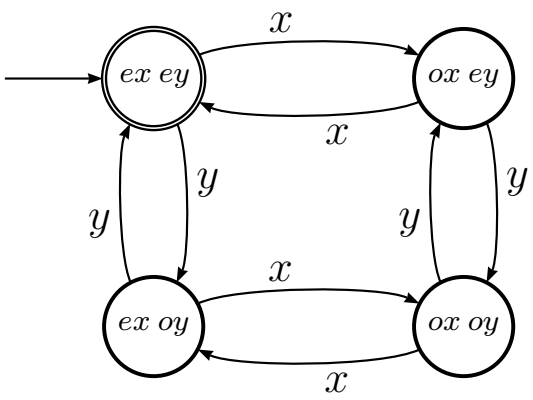
(b) Every third symbol is an x.
Answer:



Let p be the position of the symbol in the string being recognized next. The first symbol in the string has position $p = 0$.

- State s_0: $p \bmod 3 = 0$.
- State s_1: $p \bmod 3 = 1$.
- State s_2: $p \bmod 3 = 2$.

(c) All strings with an even number of x and an even number of y.
Answer:

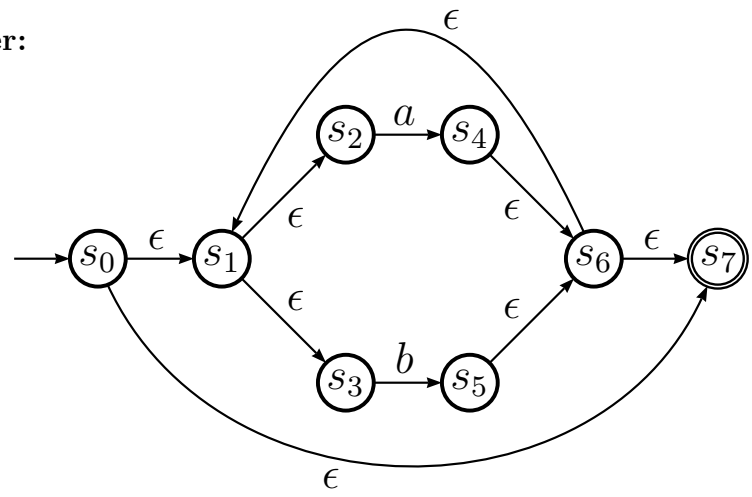


For the following state descriptions, note that 0 counts as an even number.

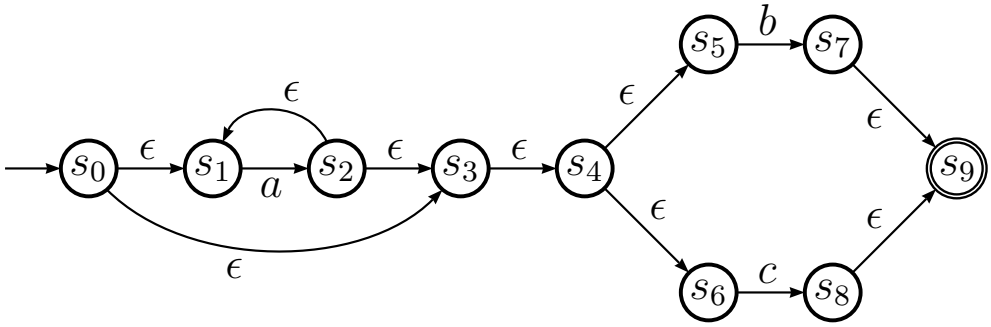
- State *ex ey*: an even number of x and an even number of y have been read.
- State *ox ey*: an odd number of x and and even number of y have been read.
- State *ex oy*: an even number of x and an odd number of y have been read.
- State *ox oy*: an odd number of x and an odd number of y have been read.

5. Use Thompson's construction algorithm to construct non-deterministic finite automata (NFAs) from the following regular expressions:

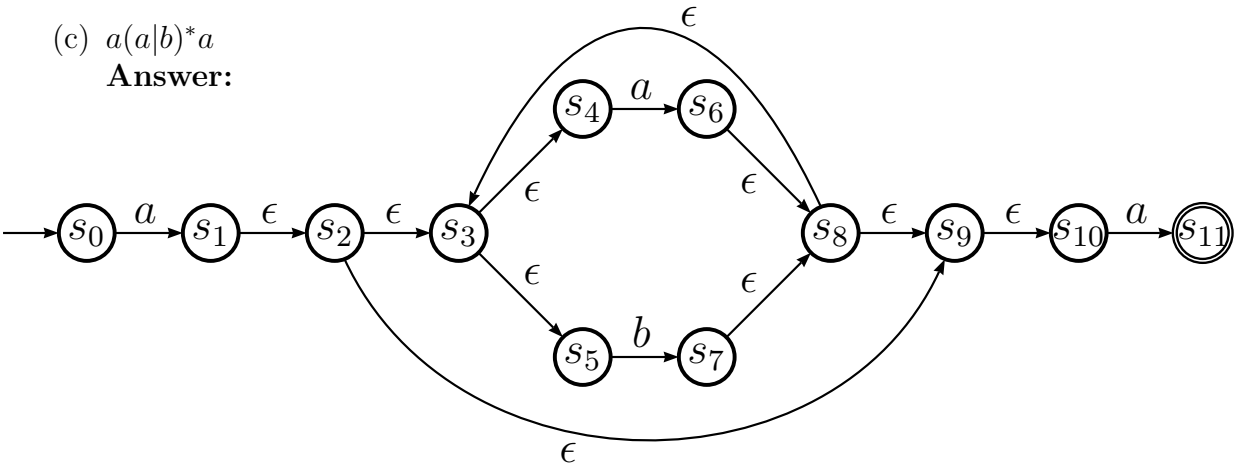
(a) $(a|b)^*$
Answer:



(b) $a^*(b|c)$
Answer:



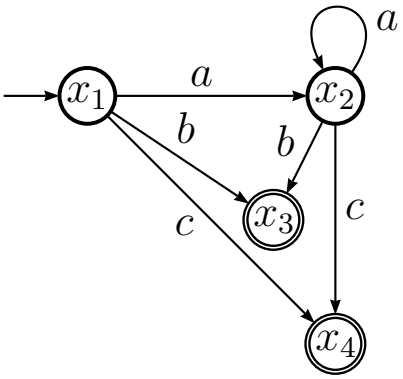
(c) $a(a|b)^*a$
Answer:



6. Use the subset construction algorithm to convert the above NFA for $a^*(b|c)$ to a DFA.

Answer:

		ϵ_closure(Move(s, *))		
DFA state	NFA states	a	b	c
x_1	$\{s_0, s_1, s_3, s_4, s_5, s_6\}$	$\{s_2, s_1, s_3, s_4, s_5, s_6\}$	$\{s_7, s_9\}$	$\{s_8, s_9\}$
x_2	$\{s_2, s_1, s_3, s_4, s_5, s_6\}$	$\{s_2, s_1, s_3, s_4, s_5, s_6\}$	$\{s_7, s_9\}$	$\{s_8, s_9\}$
x_3	$\{s_7, s_9\}$	—	—	—
x_4	$\{s_8, s_9\}$	—	—	—



Note that the above DFA is not minimal. After state minimization we get