Question 1:   NFA to DFA

The RegEx for this NFA is: x*y{1}x*z*

The NFA has 4 states: a, b, c, d. Therefore the corresponding DFA has the state set:

$\{\varnothing, \{a\}, \{b\}, \{c\}, \{d\}, \{a,b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\},$
$\{a, b, c\}, \{a, c, d\}, \{b, c, d\}, \{a, b, c, d\}\}$

The accept states for the DFA are:

$\{\{c\}, \{a, c\}, \{b, c\}, \{c, d\}, \{a, b, c\}, \{a, c, d\}, \{a, b, c, d\}\}$
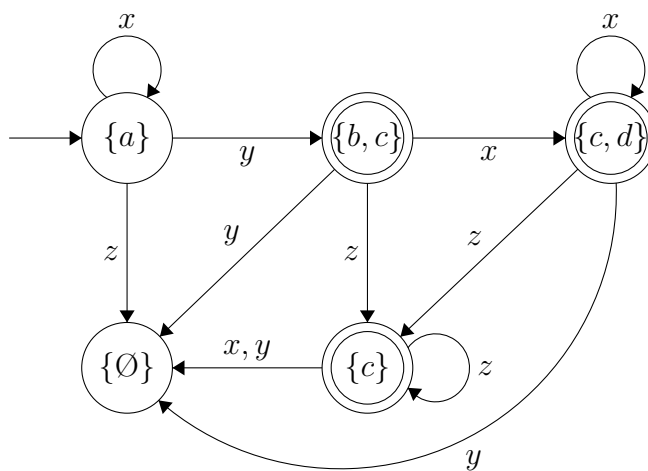
ϵ-closure(a) = {a}
ϵ-closure(b) = {b,c,d}
ϵ-closure(c) = {c}
ϵ-closure(d) = {c,d}

Transitions

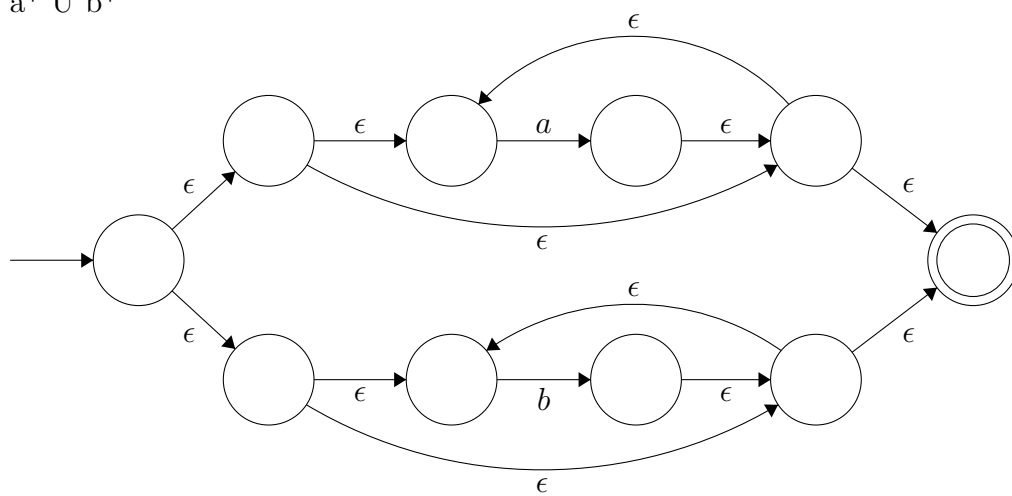| State	x	y	z
{a}	{a}	{b,c}	$\varnothing$
{b,c,d}	{c,d}	$\varnothing$	$\varnothing$
{c}	$\varnothing$	$\varnothing$	{c}
{c,d}	{c,d}	$\varnothing$	{c}

Turning all of this into a graphical representation of the corresponding DFA we get:
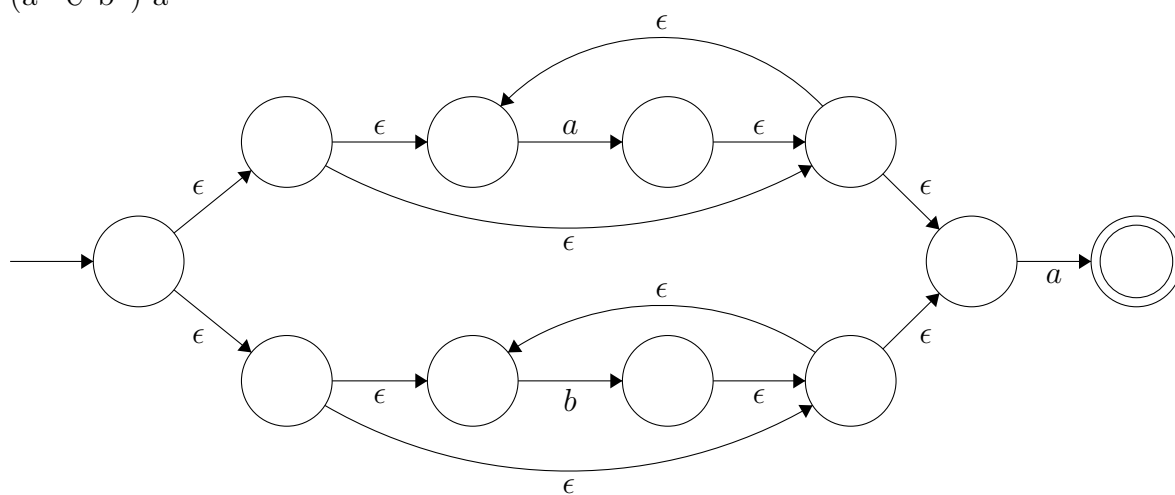
Question 2:   RegEx to NFA

a

b

a*

b*

a* U b*
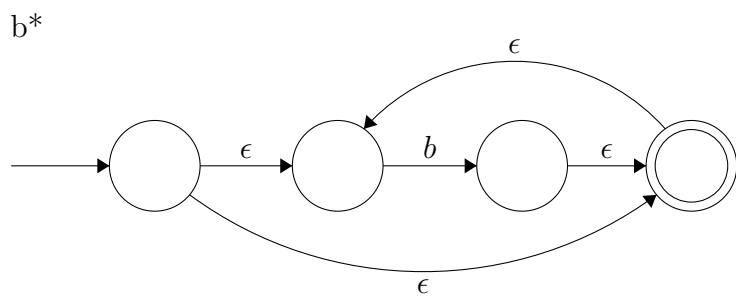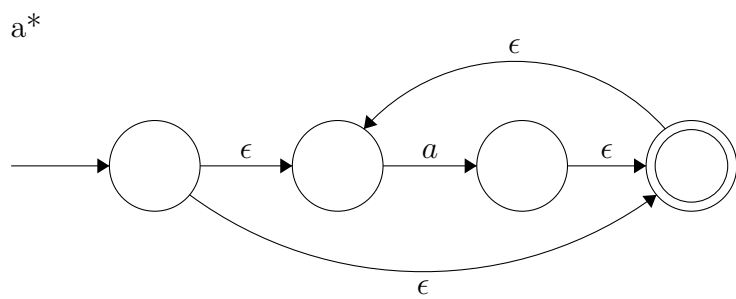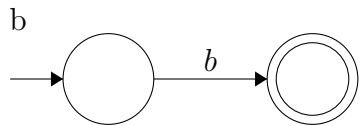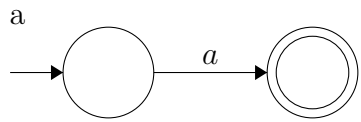
(a* U b*) a

Question 3:   Binary Additive Arithmetic

The RegEx for the Binary Additive Arithmetic language is:

$$(0? + 1(1 + 0)^*)(+|-)(0? + 1(1 + 0)^*)$$

The corresponding NFA for this language is: