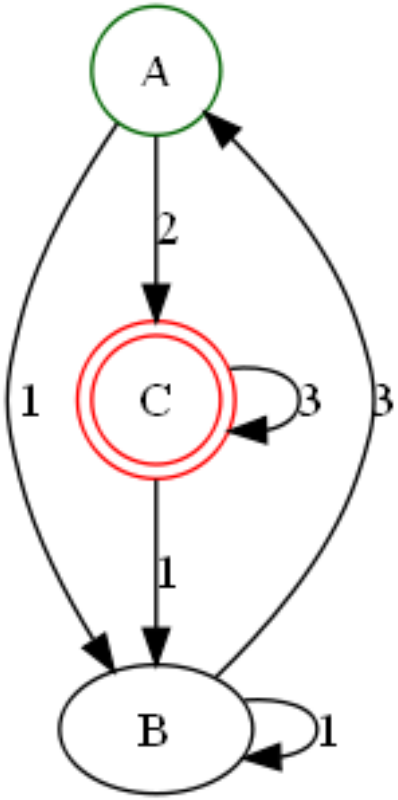
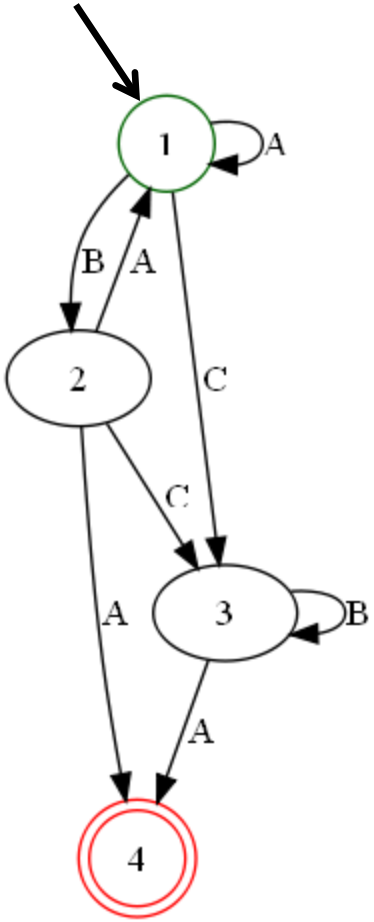
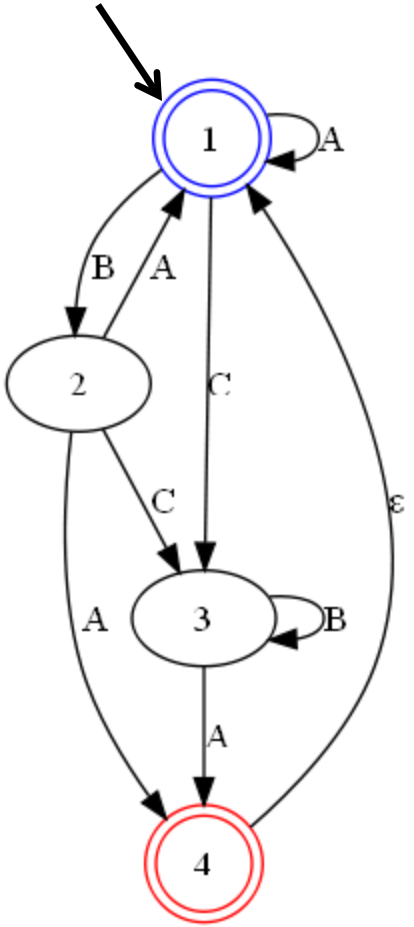
In Class Exercises

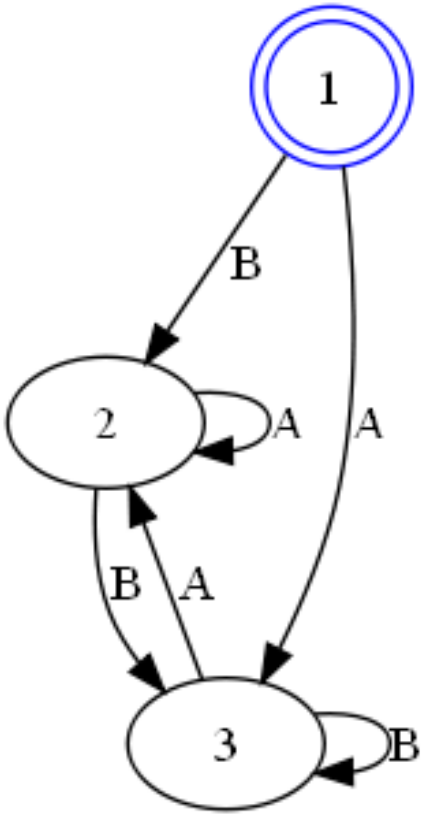


- What kind of finite state automata is this
  - DFA
  - NFA
  - GNFA





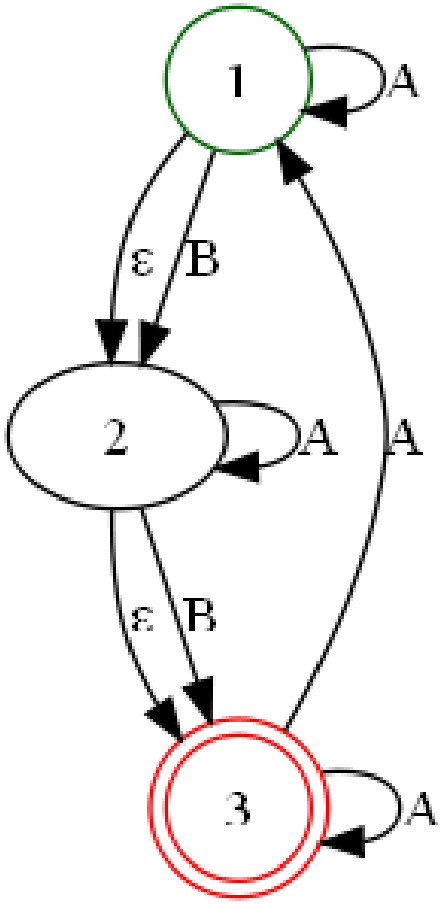
- Is the NFA on the right, the kleene closure of the NFA on the left?



- Does this DFA accept the empty language?

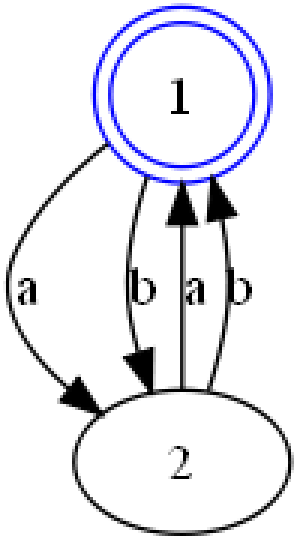
- Is the following language Context Free?
- $\{ x^i y^j z^k \mid i,j,k \geq 0, \text{ and } i + j = k \}$

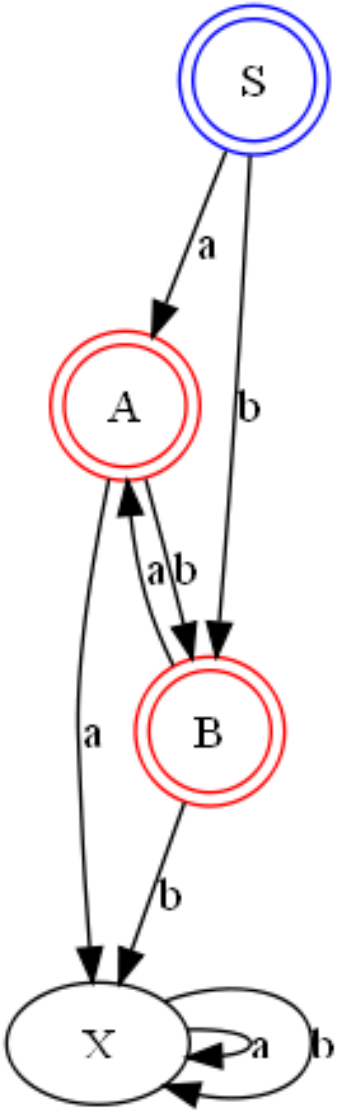
- Construct a DFA that accepts the same language as the ε-NFA below

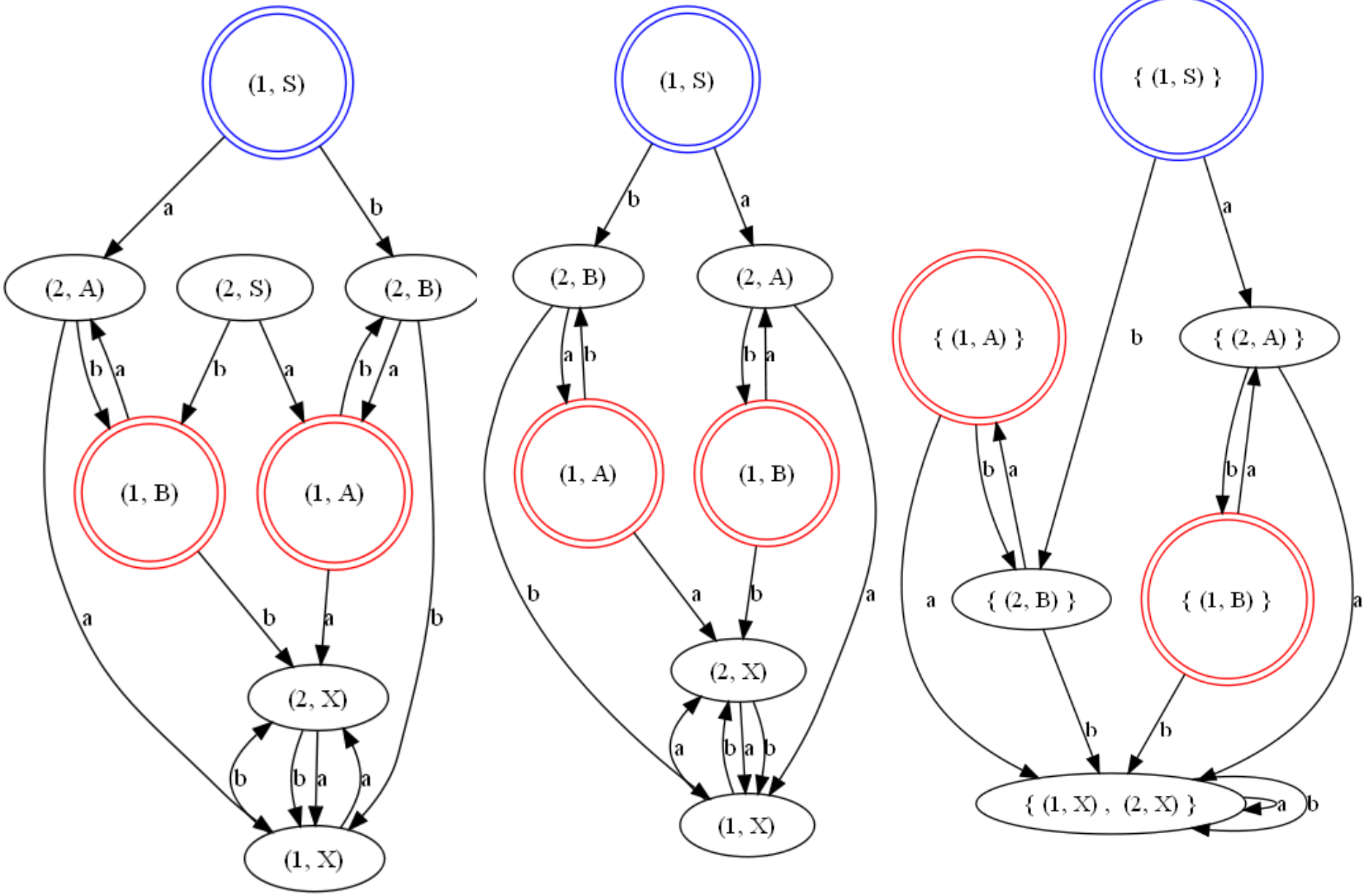


- Construct a DFA that accepts even length strings of where n two adjacent characters are the same, over the alphabet {a,b}.

1. First construct a DFA for even length strings
2. Then construct a DFA for no adjacent same characters
3. Then combine them using the correct algorithm for the right closure property of DFAs

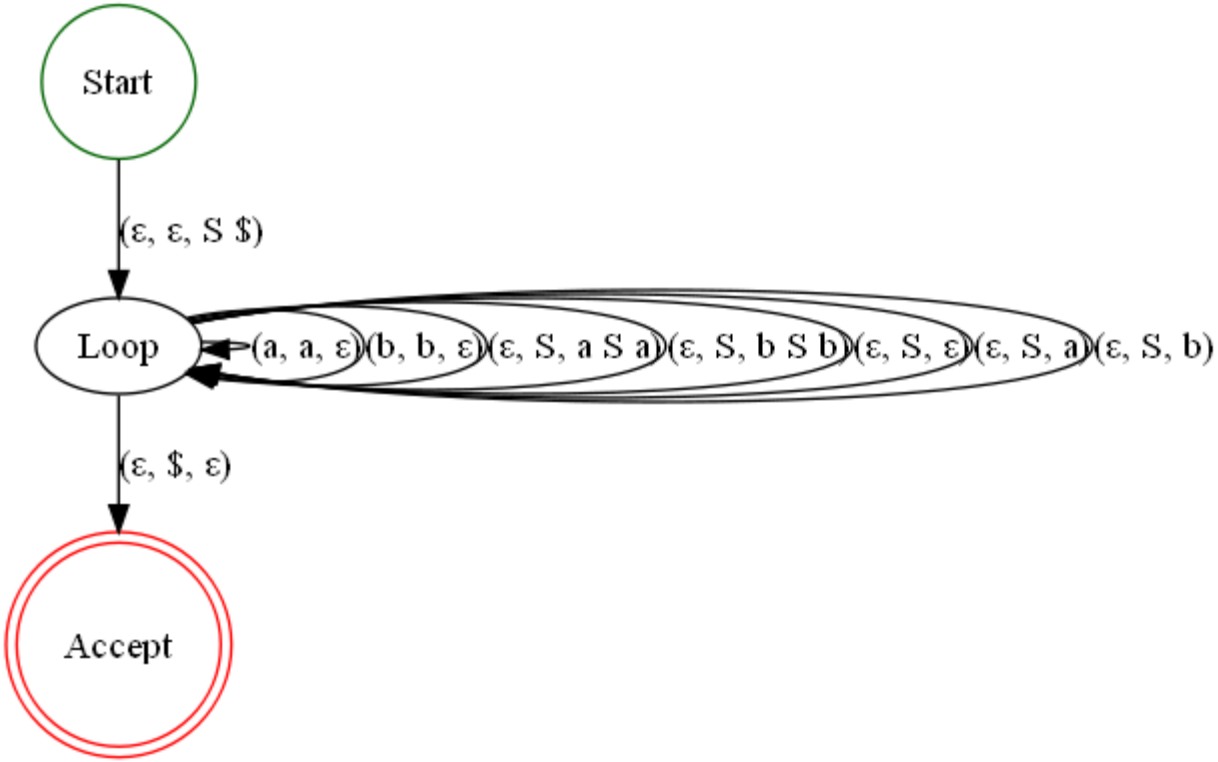

1
a
b
a
b
2


S
a
b
A
a
b
a
B
b
X
a
b


(1, S)
(2, A)
(2, S)
(2, B)
(1, B)
(1, A)
(2, X)
(1, X)
(1, S)
(2, B)
(2, A)
(1, A)
(1, B)
(2, X)
(1, X)
{ (1, S) }
{ (1, A) }
{ (2, A) }
{ (2, B) }
{ (1, B) }
{ (1, X) , (2, X) }

- Convert the CFG below into a PDA
  - It is acceptable to use a PDA that pushes strings of symbols, rather than single symbols on to the stack

```
CFGram
    NonTerm S
    Term    a b
    Start   S
    Prod    S -> a S a
            S -> b S b
            S ->
            S -> a
            S -> b
```


Start
(ε, ε, S $)
Loop
(a, a, ε)
(b, b, ε)
(ε, S, a S a)
(ε, S, b S b)
(ε, S, ε)
(ε, S, a)
(ε, S, b)
(ε, $, ε)
Accept