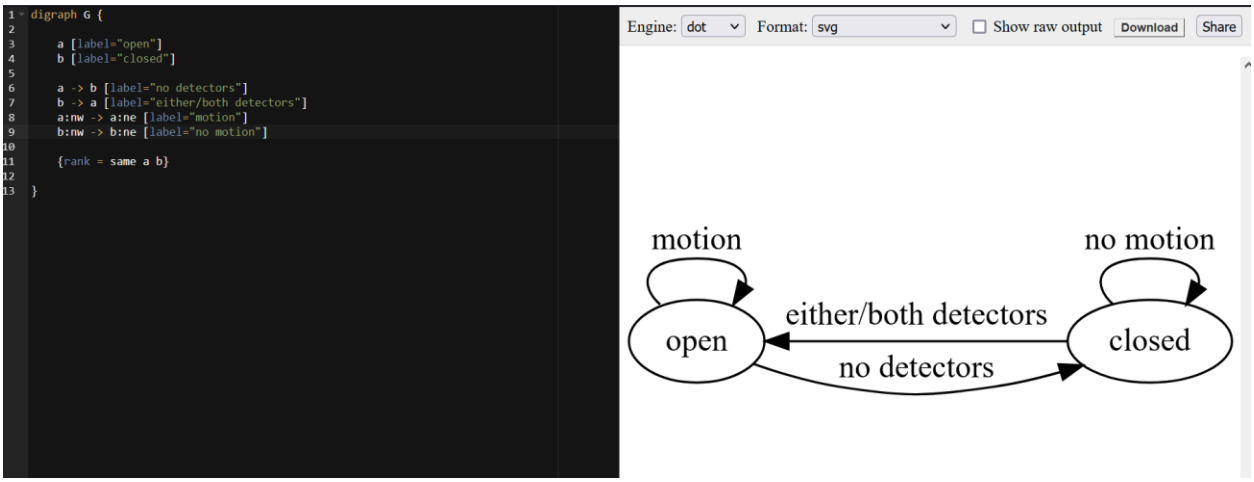
Chapter 1 Notes

Question 1:



For diagrams, circle within a circle is an accept state.

State with line from nowhere is start state

Transitions are lines

Reject states are single line circles

Question 2:

It should accept the state, as it ends on q2

Language of first diagraph (shown in class)

A = {*w* | *w* contains at least one 1 and an even number of 0s following the last 1 }

Formal Definition of finite automaton:

It is a 5-tuple (Q, Sigma, delta, q_0, F)

Q: finite set called set of states

Sigma: Finite set called alphabet

Delta: Q x Sigma -> Q is the transition function

q_0: included in Q that is the start state called initial state

F: Subset of Q, set of accept states, also called the final states

Question 3:

Q: q_1 q_2 q_3

Sigma: 0, 1

Delta: q_1: q_1 q_2

q_2: q_2 q_3

q_3: q_2

q_0: q_1

F: q_2

Question 4:

Q: q_1 q_2

Sigma: 0, 1

Delta: q_1: q_1 q_2

q_2: q_1 q_2

q_0: q_1

F: q_2

Question 5:

Q: q_0 q_1 q_2

Sigma: {0, 1, 2, *reset*}

Delta: Said not to do it

q_0: q_0

F: q_0

Sometimes the machine states are too big, so we can give a formal description instead

Q = Q_i where $Q_i = \{q_o, q_1, \ \ldots, q_{i-1}\}$

Σ = $\{< RESET >, 0,1,2\}$

δ = (next slide)

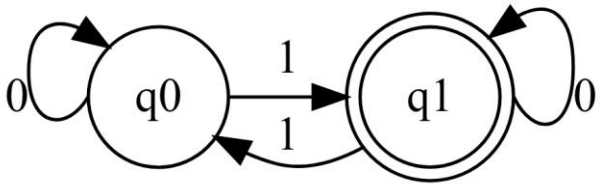
q_o = q_o

F = $\{q_o\}$

Question 6:

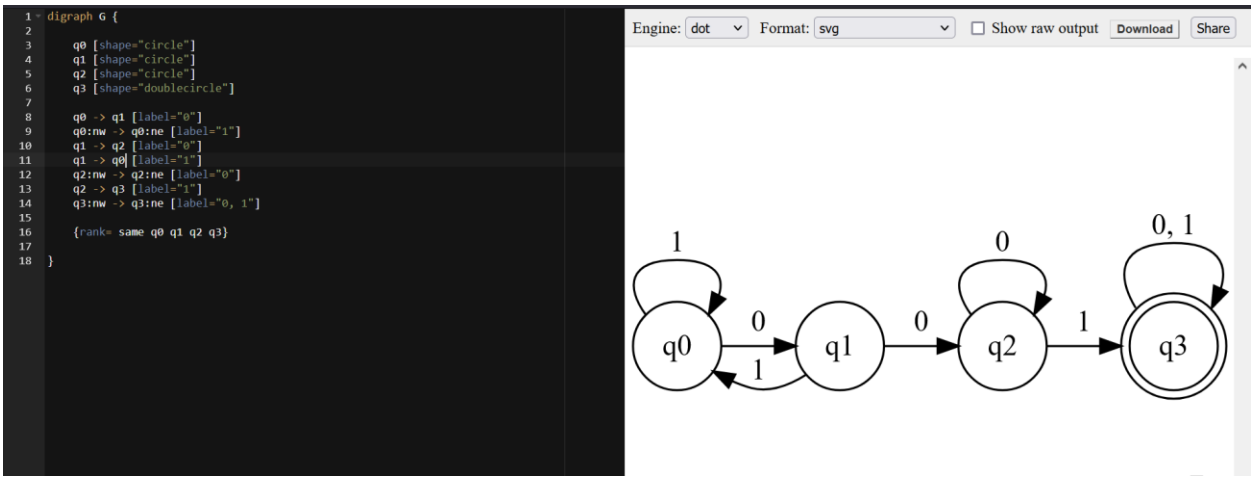
```
digraph G {

    //q(number mod 2) = 0 or 1
    //q0 represents evens and q1 represents odds

    a [shape=circle, label="q0"]
    b [shape=doublecircle, label="q1"]

    a -> b [label="1"]
    b -> a [label="1"]
    b:e -> b:ne [label="0"]
    a:w -> a:nw [label="0"]

  {rank = same a b}

}
```

Engine: dot Format: svg Show raw output Download Share



With the above, you can change the logic to be an even number of 1s by switching the accept state.

Question 7:



Objects are our languages, and the tools for language manipulation are Union, Concatenation, and Star

Question 8:

AuB = {good, bad, boy, girl}

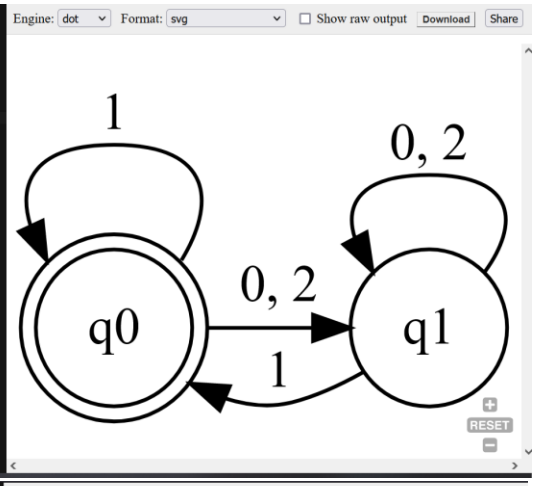
A o B = {good, boy} {good, girl} {bad, boy} {bad, girl}

A* = { empty, good, bad, good good, bad bad, good bad, bad good, bad bad, good good good, good good bad, …}

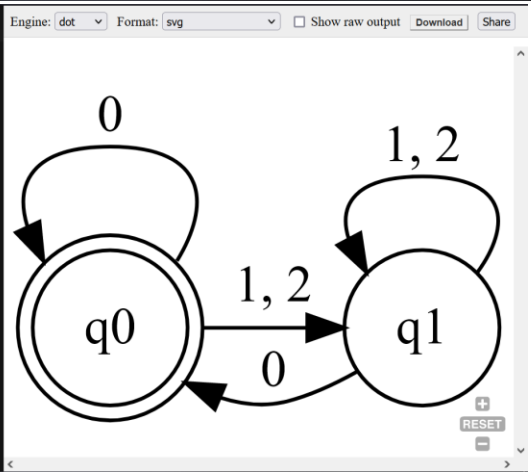
Much like math operators will output a number, these will output language if language is inputted.

Question 9:

```
digraph G {

    q0 [shape="doublecircle"]
    q1 [shape="circle"]


    q0:ne -> q0:nw [label="1"]
    q1 -> q0 [label="1"]
    q0 -> q1 [label="0, 2"]
    q1:ne -> q1:nw [label="0, 2"]


    {rank=same q0 q1}
}
```




```
digraph G {

    q0 [shape="doublecircle"]
    q1 [shape="circle"]


    q0:ne -> q0:nw [label="0"]
    q1 -> q0 [label="0"]
    q0 -> q1 [label="1, 2"]
    q1:ne -> q1:nw [label="1, 2"]


    {rank=same q0 q1}
}
```




```
digraph G {

    q0 [shape="doublecircle"]
    q1 [shape="circle"]


    q0:ne -> q0:nw [label="(0, 1)"]
    q1 -> q0 [label="(0,1)"]
    q0 -> q1 [label="(0; 2, 1; 2)"]
    q1:ne -> q1:nw [label="(0; 2, 1; 2)"]


    {rank=same q0 q1}
}
```