**Problem #1**

a.

Not Reflexive => {1,1} not in R

Symmetric => {1,3} has {3,1} and (2,2) is inherently symmetric

Not Transitive => {1,3} & {3,1} but not {1,1}

b.

Reflexive => {1,1}, {2,2} & {3,3} in R

Not Symmetric => {1,2} in R but {2,1} is not

Transitive => for all a, b, c in S if (a,b) & (b,c) -> (a,c)

c.

Not Reflexive => {1,1} not in R

Symmetric => no {a,b} in R so P is false, so p -> q is true

Transitive => no {a,b} in R so P is false, so p -> q is true

d.

{2,1} would need to be added to make R an eq rel

**Problem #2**

a.

Base:

If w=ε concatenate with the string ‘a’

Recursion:

Case 1:

concatenate a or b to left side of w

Case 2:

concatenate a or b to right side of w

Derivation:

a -> base

ab -> rec, case 2

aba -> rec, case 2

abab -> rec, case 2

b.

Base:

If w=ε concatenate with 00

then, concatenate 1 or 0 to left side of w

Recursion:

Concatenate w with any binary string of length 2 on the left side

Derivation:

00 -> base part 1

100 -> base part 2

10100 -> rec

**Problem #3**

**Problem #4**

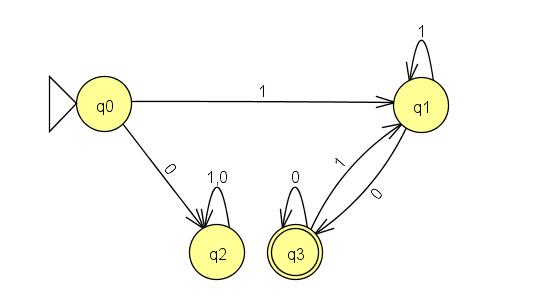
a.

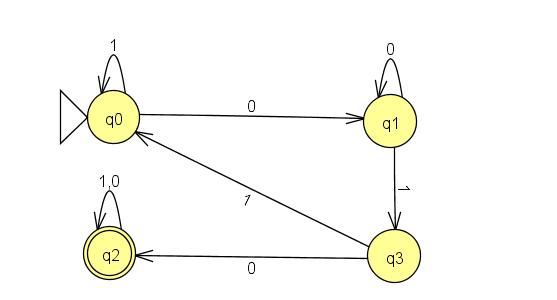
L = a\*

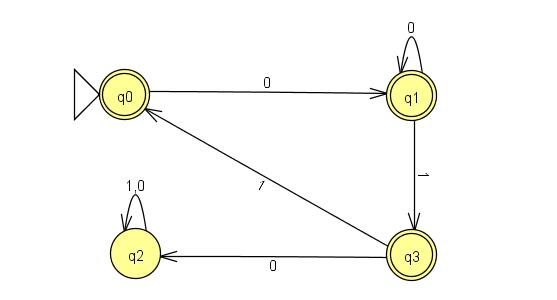
b.

L = {w | there are an even number of ‘a’s and ‘b’s}

**Problem #5**







**Problem #6 ?**

1.

Not Reflexive => (5,5) is not an element of R

Symmetric => by commutative property of addition, x+y = y+x

Not Antisymmetric => (3, -3) != (-3, 3)\

Transitive

**Problem #7**

1.

This is an eq rel

2.

Not an eq rel

Not Transitive => (0,2) & (2,3) but not (0,3)

3.

This is an eq rel

4.

Not an eq rel

Not Symmetric => (1,2) in R but not (2,1)

**Problem #8**

Reflexive

(a,b)R(a,b) -> a+b = b+a

True by commutative property of addition

Not Symmetric

(2,3) != (3,2)

Transitive

(a,b)R(c,d) ^ (c,d)R(e,f) -> a+b=b+c and b+c = d+e

a+b=c+d=d+e

So, (a,b)R(e,f)