Problem 1

• bowling balls are sporting equipment
∀x BowlingBall(x) -> sportingEquipment(X)

• all domesticated horses have an owner
∀x∃y horse(x) ^ person(y) ^ owner(y, x)

• the rider of a horse can be different than the owner
∃x∃y∃z horse(x) ^ person(y) ^ person(z) ^ owner(y, x) ^ rides(z, x)

• horses move faster than frogs
∀x, y horse(x) ^ frog(y) -> faster(x, y)

• a finger is any digit on a hand other than the thumb
∀x finger(x) ^ notThumb(x) -> digit(x)

• an isosceles triangle is defined as a polygon with 3 edges, which are connected at 3 vertices, where 2 (but not 3) edges have the same length

∀x,y,x sidesOfTriangle(x,y,z) ^ areConnected(x,y,z) ^ ((equal(x,y)^notEqual(y,z)) v (equal(y,z) ^ notEqual(z,x)) v (equal(z,x) ^ notEqual(x,y))) -> FormIsosceles(x,y,z)

Problem 2

∀ x person(x) ^ [∃z petOf(x,z) ^ ∀y petOf(x,y)→dog(y)] → doglover(x)

∀x person(x) ^ [∃z petOf(x, z) ^ ∀y petOf(x, y) -> dog(y)] -> doglover(x, y)

For my purposes: The above is saying for all people, if they own a pet and if they only own pets that are dogs, then they are dog lovers

$\forall x\ person(x) \wedge [\exists z\ petOf(x,z) \wedge \forall y\ petOf(x,y) \rightarrow dog(y)] \longrightarrow doglover(x)$ Implication Elimination

$\forall x\ \neg(person(x) \wedge Ez\ petOf(x,z) \wedge \forall y\ petOf(x,y) \rightarrow dog(y)) \vee doglover(x)$ simplify / move in negation

$\neg(A \rightarrow B) = A \wedge \neg B$

$\neg(person(x) \wedge Ez\ petOf(x,z) \wedge \forall y\ petOf(x,y) \wedge \neg dog(y)) \vee doglover(x)$

do universal substitution and skolemization

$\neg(person(x) \wedge petOf(x, f(x)) \wedge petOf(x,y) \wedge \neg dog(y)) \vee doglover(x)$

- push in negation

$(\neg person(x) \vee \neg petOf(x, f(x)) \vee \neg petOf(x,y) \vee dog\ y) \vee doglover(x)$

Problem 3

3. Determine whether or not the following pairs of predicates are unifiable. If they are, give the
most-general unifier and show the result of applying the substitution to each predicate. If they are
not unifiable, indicate why. Capital letters represent variables; constants and function names are
lowercase. For example, ‘loves(A,hay)’ and ‘loves(horse,hay)’ are unifiable, the unifier is u={A/horse}, and the unified expression is ‘loves(horse,hay)’ for both.

• owes(owner(X), citibank, cost(X)) owes(owner(ferrari), Z, cost(Y))
U = {ferrari/X,ferarri/Y, citibank/Z}
owes(owner(ferarri, citibank,ferrari)

• gives(bill, jerry, book21) gives(X, brother(X),Z)
u={bill/X, brother(X)/jerry, book21/Z}

gives(bill, jerry, book21)

• opened(X, result(open(X),s0))) opened(toolbox, Z)
Not unifiable

- Arguments of result are 2, not sure what to reference to get it unifiable

Problem 4. (On Paper)

Problem 4
Part a

1.

Marcus is pompeian (1).

$pompeaian(Marcus)$.

All pompeians are Roman. (2)

$\forall x \; pompeaian(x) \rightarrow Roman(x)$.

Ceaser is a ruler (3).

$Ruler(Ceaser)$.

All Romans are loyal to or hate Ceaser (but not both). (4)

$\forall x \; Roman(x) \longrightarrow (Loyal(x, Ceasar) \wedge \neg hate(x, Ceasar)) \vee$
$(\neg loyal(x, Ceasar) \wedge hate(x, Ceasar))$.

Everyone is loyal to someone (5)

$\forall x \, \exists y \; loyal(x, y)$.

people only assinate rulers they are not loyal to. (6)

~~∀x,y assinate(x,y) → ¬loyal(x,y)~~

Marcus Tries to assinate Ceaser (7).

$assasinate(Marcus, Ceasar)$.

$\forall x,y, \; assinate(x,y) \longrightarrow Ruler(y) \wedge \neg loyal(x,y)$

part b

1. Use universal instantiation on (7) and (6). to get:

   1 assassinate (Marcus, Ceasar) → Ruler (Ceasar) ∧ ¬ Loyal (Marcus, Ceasar)

2. Since assinate (marcus, Ceasar) ∧
   assinate (Marcus, Ceasar) → Ruler (Ceasar) ∧ ¬ loyal (Marcus, Ceasar)

   we know ¬ Loyal (Marcus, Ceasar)

3. Marcus is Roman, or Roman Marcus (Rules 1,2)

4. Roman (Marcus) → (Loyal (Marcus Ceasar) ∧ ¬ hate (Marcus, Ceasar) ∨
   ¬ Loyal (Marcus, Ceasar) ∧ hate (Marcus, Ceasar))

5. This portion has to be true, since we know ¬ loyal (Marcus, Ceasar)

   So hate (Marcus, Ceasar).

Part c

3.

pompeain (Marcus).

pompeain (x) → Roman (x) Getting Rid of Quantifiers.

Ruler(Ceasar)

Roman(x) → ((loyal(x, Ceasar) ^ ¬ hate(x, Ceasar)) V (¬loyal(x, Ceasar) ^ hate(x, Ceasar)

u.

Loyal(x, f(x))

3.

assasinate(x,y) → Ruler(y) ^ ¬ loyal(x,y).

assasinate(marcus, Ceasar).

need to get rid of these implications now.

1. ¬ pompean (x) V Roman (x).
2. ¬ Roman (x) V (u)
3. ¬ assasinate (x,y) V (Ruler(y) ^ ¬ loyal (x,y))

-simplify 2 and 3.

Problem 5

∃ ∀

MapColoring

KB = {

∀x state(x) -> ∃y hasColor(x, y)

∀x,y,z state(x) ^ color(y) ^ color(z) ^hasColor(x,y) ^hasColor(x,z) -> y=z

∀x,y,z state(x) ^ state(y) ^ color(z) ^ hasColor(x,z) -> ¬hasColor(y,z)

}

Wumpus World ∃ ∀
KB:
{
∀x block(x) ^ stench(x) -> ∃y block(y) ^ Wumpus(y) ^ adj(x,y)
∀x block(x) ^ breeze(x) -> ∃y block(y) ^ pit(y) ^ adj(x,y)
}

Sammy's sports shop
These facts were supplied in the original homework 2 for sammy's sports shop

: {O1Y, L1W, O2W, L2Y, O3Y, L3B}
Converting to FOL
Y, yellow, w, white, b, both
1, box1, 2, box2, 3, box3

obs(1,y), labeled(1,w), obs(2,y), obs(3,y), labeled(3,both)

KB:
{∀x,y box(x) ^ color(y) ^ labeled(x,y) -> ¬ contains(x,y)
∀x,y,z box(x) ^ obs(x,y) ^ obs(x,z) ^ ¬ (z=y) -> contains(x,both)
∀x,y,z,a box(x) ^ box(y) ^ box(z) ^ color(a) ^ obs(x,a) ^ obs(y,a) -> ¬ contains(z,a)
∀x,y,z,a box(x) ^ box(y) ^ obs(x,z) ^ obs(x,a) -> ¬ contains(z,both)
∀x,y,z,a box(x) ^ box(y) ^ obs(x,z) ^ obs(x,a) -> ¬ contains(z,both)
}

4-queens
KB = {
∀x,y queen(x) ^ queen(y) -> ¬ (col(x) = col(y)) ^ ¬(row(x) = row(y))
∀x,y queen(x) ^ queen(y) -> ¬ ((xDistance(row(x), row(y)) = yDistance(row(x), row(y))
}