CSC 380/480  
A3

Q1

* knight(A) → ⌐ knave(A)
* knave(A) → ⌐ knight(A)
* says(a,s) ʌ knight(A) → s
* says(a,s) ʌ knave(a) → ⌐ s
* Suppose knight(a)
* Therefore says(a,”b is a naïve”)) ʌ knight(A) → “b is a naïve”
* So knave(b)
* Therefore says(a,”neither a nor I are knaves”) ʌ knave(a) → ⌐ ”neither a nor I are knaves”
* So knight(a) and knave(b)

Q2:

* knight(A) → ⌐ knave(A)
* knave(A) → ⌐ knight(A)
* says(a,s) ʌ knight(A) → s
* says(a,s) ʌ knave(a) → ⌐ s
* Suppose knave(a)
* Therefore says(a,”we are the same kind”) ʌ knave(a) → ⌐ ”we are the same kind”
* So knight(b)
* Therefore says(a,”we are different kinds”)) ʌ knight(A) → “we are different kinds”
* So knight(b) and knave(a)

Q3:

* knight(A) → ⌐ knave(A)
* knave(A) → ⌐ knight(A)
* says(a,s) ʌ knight(A) → s
* says(a,s) ʌ knave(a) → ⌐ s
* r(1) → death
* r(2) → freedom
* Suppose A is guarding r(2).
* Suppose B is guarding r(1).
* Suppose we point to R(2) and ask the person ‘will the other person this path is correct?”
* Says(a, no), as b will lie.
* Says(b, no) as the knight would say it’s the correct path and therefore the naïve must lie.
* Therefore, we can prove we are pointing to the proper path.

Q4-Q5.

* Creatures can clobber any creature they eat.

*Let c be creature*

*Let e be eat*

*Let clob be clober*

*Initial State*

∀x ∃y [ c(x) ʌ c(y) ] ʌ e(x,y) → clob(x,y)

*Simplify*

∀x ∃y ⌐{ [ c(x) ʌ c(y) ] ʌ e(x,y) } v clob(x,y)

*Move Negations In*

∀x ∃y { ⌐[ c(x) ʌ c(y) ] v e(x,y) } v clob(x,y)

∀x ∃y { [ ⌐c(x) v ⌐c(y) ] ʌ ⌐ e(x,y) } v clob(x,y)

*Distribute Qualifiers*

∀x ∃y [ [ ⌐c(x) v ⌐c(y) ] ʌ ⌐ e(x,y) v clob(x,y) ]

*Skolemization*

[ [ ⌐c(**λ{x}**) v ⌐c(**x{λ}**) ] ʌ ⌐e(**λ{x}**, **x{λ}**) v clob(**λ{x}**, **x{λ}**) ]

*Distribute Disjunctions*

{ [ ⌐c(**x**) v ⌐c(x{**λ}**) ] v clob(**λ{x}**, **x{λ}**) } ʌ { ⌐e(**x**,x{**λ})** v clob(**x**,x{**λ}**) }

*Convert to CNF*

{ [ ⌐c(**x**) v ⌐c(x{**λ}**) ] v clob(**x**, **λ{x}**) } , { ⌐e(**x**,x{**λ})** v clob(**x**,x{**λ}**) }

* Monsters eat some other creatures.

*Let c be creature*

*Let e be eat*

*Let m be monster*

*Initial State*

∀x ∃y (m(x) ʌ c(y) → e(x,y))

*Simplify*

∀x ∃y ⌐(m(x) ʌ c(y) v e(x,y))

*Move Negations In*

∀x ∃y (⌐m(x) v ⌐c(y) v e(x,y))

*Distribute Qualifiers*

∀x ∃y [ (⌐m(x) v ⌐c(y) v e(x,y)) ]

*Skolemization*

[ (⌐m(x) v ⌐c(x{**λ}**) v e(x, x{**λ}**)) ]

*Distribute Disjunctions*

[ ⌐m(x) v e(x, x{**λ}**) ] ʌ [ ⌐c(x{**λ}**) v e(x, x{**λ}**) ]

*Convert to CNF*

{ ⌐m(x) v e(x, x{**λ}**) } , { ⌐c(x{**λ}**) v e(x, x{**λ}**) }

* Clobbering is transitive, i.e., if x clobbers y, and y clobbers z, then x clobbers z.

*Let clob be clobbering*

*Let p be person*

*Initial State*

*Simplify*

*Move Negations In*

*Distribute Qualifiers*

*Skolemization*

*Distribute Disjunctions*

*Convert to CNF*

* Ogres eat dwarves.

*Let o be ogres*

*Let d be dwarf*

*Let e be eat*

*Initial State*

∀x ∀y (o(x) ʌ d(y)) → e(x,y)

*Simplify*

∀x ∀y ⌐(o(x) ʌ d(y)) v e(x,y)

*Move Negations In*

∀x ∀y (⌐o(x) v ⌐d(y)) v e(x,y)

*Distribute Qualifiers*

∀x ∀y [ (⌐o(x) v ⌐d(y)) v e(x,y) ]

*Skolemization*

[ (⌐o(x) v ⌐d(x{**λ}**)) v e(x, x{**λ}**) ]

*Distribute Disjunctions*

{ ⌐o(x) v e(x, x{**λ}**) } ʌ {⌐d(x{**λ}**) v e(x, x{**λ}**)}

*Convert to CNF*

{ ⌐o(x) v e(x, x{**λ}**) } , {⌐d(x{**λ}**) v e(x, x{**λ}**)}

* Dwarves can clobber goblins.

*Initial State*

∀x ∀y (d(x) ʌ g(y)) → clob(x,y)

*Simplify*

∀x ∀y ⌐(d(x) ʌ g(y)) v clob(x,y)

*Move Negations In*

∀x ∀y (⌐d(x) v ⌐g(y)) v clob(x,y)

*Distribute Qualifiers*

∀x ∀y [ (⌐d(x) v ⌐g(y)) v clob(x,y) ]

*Skolemization*

[ (⌐d(x) v ⌐g(x{**λ}**)) v clob(x, x{**λ}**) ]

*Distribute Disjunctions*

{ ⌐d(x) v clob(x, x{**λ}**) } ʌ {⌐g(x{**λ}**) v clob(x, x{**λ}**)}

*Convert to CNF*

{ ⌐d(x) v clob(x, x{**λ}**) } , {⌐g(x{**λ}**) v clob(x, x{**λ}**)}

* Goblins are monsters.

*Let g be goblin*

*Let m be monster*

*Initial State*

*Simplify*

*Move Negations In*

*Distribute Qualifiers*

*Skolemization*

*Distribute Disjunctions*

*Convert to CNF*

**Q6. Use resolution-refutation to prove that an ogre can clobber a goblin. In addition to the facts explicitly stated above, you may need to write other statements to do the proof.**

**Q7. What other creatures can an Ogre eat? Show your work.**

Q8a

* True

Q8b

* x = mia

Q8c

* x = mia

Q8d

* false

Q8e

* x = Vincent, x = marsellus

Q8f

* x = Vincent, x = marsellus

Q8g

* x = honey\_bunny

Q9.

* is\_car(bmw).
* is\_car(civic).
* is\_moto(harley).
* is\_fast(bmw).
* is\_fast(harley).
* is\_fast(flash).
* is\_slow(civic).
* is\_fun(A) :- is\_car(A),is\_fast(A).
* is\_fun(A) :- is\_moto(A),is\_fast(A).
* is\_fun(bmw).
  + True
* is\_fun(flash).
  + False
* is\_fun(civic).
  + False
* is\_fun(harley).
  + True

Q10:

1. Ogre cannot clobber a goblin.
2. Orges eat dwarfs.

* ogres(x).
* dwarf(y).
* goblin(z).
* eat(A,B) :- ogres(A),dwarf(B).
* clobber(A,B) :- dwarf(A),goblin(B).
* clobber(x,z)
  + True
* eat(x,y)
  + True