COMP 681 Formal Methods Spring 2008 Assignment 5—<u>Solutions</u>

1. You are given the following predicates.

child(*x*, *y*) means *x is a child of y*
sibling(*x*, *y*) means *x is a sibling of y*
female(*x*) means *x is a female*
married_to(*x, y*) means *x is married to y*

Define the following predicate in terms of these. Note that there are two ways *x* can be the aunt of *y*.

aunt(*x, y*) means *x is the aunt of y*

Answer

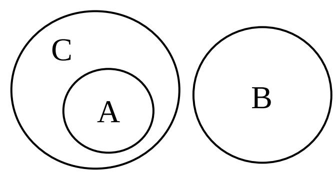
$female(x) \wedge sibling(z, x) \wedge child(y, z) \vee$
$female(x) \wedge married\_to(x, u) \wedge sibling(u, w) \wedge child(y, w)$

2. Use Venn diagrams to evaluate the validity of the following syllogisms. If the syllogism is valid, explain why you cannot draw a diagram in which the premises are true yet the conclusion is false.

a. No *A*'s are *B*'s.
<u>No *C*'s are *B*'s.</u>
Some *A*'s are not *C*'s.

Answer

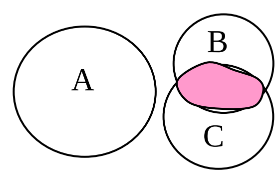
Invalid. The diagram at right satisfies the premises but not the conclusion.



b. No A's are B's.
<u>Some C's are B's</u>.
Some C's are not A's.

Answer

Valid. Referring to the diagram at right, the part of the C's that are B's (shaded) also cannot be A's. So some C's cannot be A's.



3. Translate the following argument into the language of predicate logic and use a Venn diagram to show that it is valid. Be sure to indicate the meanings of the predicate symbols you use, and <u>explain</u> why the argument is valid.

Some cars and some trucks administered by the University are state-owned and some are owned by the university. Everything owned by the University is bargain-priced. Bargain-priced vehicles are dangerous. So, since cars and trucks are vehicles, the University owns some dangerous things.

Answer

Meanings of predicates:

car(x) means *x is a car*. C denotes the set of cars.

truck(x) means *x is a truck*. T denotes the set trucks.

admin_u(x) means *x is administered by the University*. U denotes the set of things administered by the University.

state_owned(x) means *x is state-owned*. S denotes the set of state-owned things.

u_owned(x) means *x is owned by the University*. U denotes the set of things owned by the university.

bargain_priced(x) means *x is bargain-priced*. B denotes the set of bargain-priced things.

vehicle(x) means *x is a vehicle*. V denotes the set of vehicles.

dangerous(x) means *x is dangerous*. D denotes the set of dangerous things.

The following is the encoding of the above argument. We have encoded the first statement as four premises. As regards the logical consequences, this is equivalent to encoding it as a single premise that is the conjunction of the four given.

$\exists\, x \bullet car(x) \wedge admin\_u(x) \wedge state\_owned(x)$

$\exists\, x \bullet car(x) \wedge admin\_u(x) \wedge u\_owned(x)$

$\exists\, x \bullet truck(x) \wedge admin\_u(x) \wedge state\_owned(x)$

$\exists\, x \bullet truck(x) \wedge admin\_u(x) \wedge u\_owned(x)$

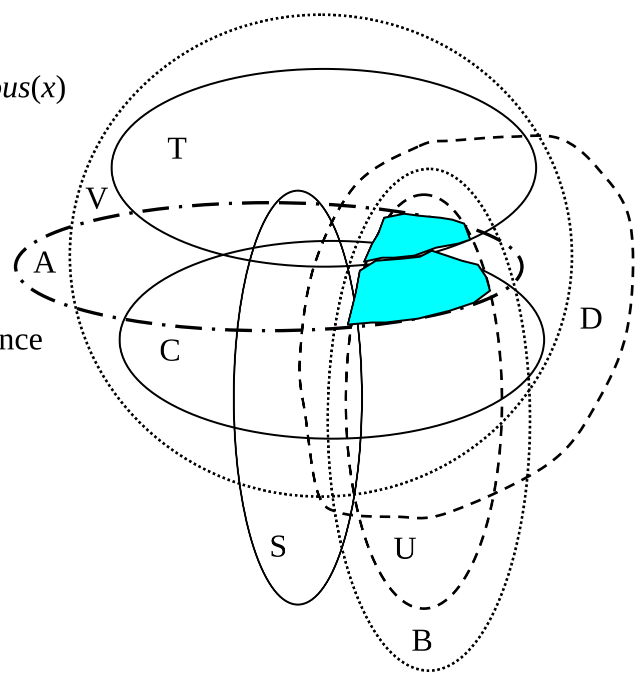
$\forall\, x \bullet u\_owned(x) \Rightarrow bargain\_priced(x)$

$\forall\, x \bullet bargain\_priced(x) \wedge vehicle(x) \Rightarrow dangerous(x)$

$\forall\, x \bullet car(x) \vee truck(x) \Rightarrow vehicle(x)$

Therefore, $\exists\, x \bullet u\_owned(x) \wedge dangerous(x)$

The diagram at right shows that this is <u>valid</u>. The set of cars and trucks administered and owned by the University, which we know is non-empty, is shaded. These are bargain-priced vehicles and hence are dangerous.



4. Encode the following statements in the language of predicate logic using the indicated predicates. Be sure to indicate the meanings of the predicate symbols you use. The encodings do not involve nested quantifiers.

a. *Old dogs and tame cats chase mice but don't bother people as long as they're cared for*.

Let

old(x) mean *x is old*
dog(x) mean *x is a dog*
tame(x) mean *x is tame*
cat(x) mean *x is a cat*
chases_mice(x) mean *x chases mice*
bother_people(x) mean *x bothers people*
cared_for(x) mean *x is cared for*

Answer

$\forall\ x \bullet old(x) \land dog(x) \lor tame(x) \land cat(x)$
$\quad \Rightarrow chases\_mice(x) \land (\ cared\_for(x) \Rightarrow \neg\ bother\_people(x)\ )$

b. *Some beer is healthy when drunk sparingly and some is unhealthy when drunk by pregnant women or old men*.

Let

beer(x) mean *x is beer*
healthy(x) mean *x is healthy*
drunk_sparingly(x) mean *x is drunk sparingly*
drunk_by_pgs(x) mean *x is drunk by pregnant women*
drunk_by_om(x) mean *x is drunk by old men*

Answer

$\exists\ x \bullet beer(x) \land (\ drunk\_sparingly(x) \Rightarrow healthy(x)\ )$
$\land\ \exists\ x \bullet beer(x) \land (\ drunk\_by\_pgs(x) \lor drunk\_by\_om(x) \Rightarrow \neg\ healthy(x)\ )$

5. Write abstract programs that specify the following.

a. *As long as x has a niece or a nephew, make y equal to a niece of x or, if x has no niece, make y equal to a nephew of x.*

Use

$sister(x, y) = x$ *is a sister of* y.

$brother(x, y) = x$ *is a brother of* y.

$son(x, y) = x$ *is a son of* y.

$daughter(x, y) = x$ *is a daughter of* y.

Answer

$$y : [\ \exists z \bullet (\ brother(z, x) \vee sister(z, x)\ ) \wedge \exists u \bullet son(u, z) \vee daughter(u, z),$$
$$(\ \exists z \bullet (\ brother(z, x) \vee sister(z, x)\ ) \wedge (\ \exists u \bullet daughter(u, z)\ )$$
$$\Rightarrow \exists z \bullet (\ brother(z, x) \vee sister(z, x)\ ) \wedge daughter(y, z)\ )$$
$$\wedge\ (\ \neg\ (\ \exists z \bullet (\ brother(z, x) \vee sister(z, x)\ ) \wedge (\ \exists u \bullet daughter(u, z)\ )\ )$$
$$\Rightarrow \exists z \bullet (\ brother(z, x) \vee sister(z, x)\ ) \wedge son(y, z)\ )\ ]$$

b. Consider a line

$$y = m*x + b$$

and a circle

$$(x-x_0)^2 + (y-y_0)^2 = r^2$$

As long as the line and the circle intersect, make the values of u and v such that (u,v) is that point of intersection of the line and the circle that is closest to the origin. (Recall that a line and a circle, if they intersect, intersect at one or two points.) Treat m, b, x_0, y_0, and r as global (read-only) variables.

Answer

$$u, v : [\ \exists x, y \bullet y = m*x + b\ \wedge\ (x-x_0)^2 + (y-y_0)^2 = r^2,$$
$$v = m*u + b\ \wedge\ (u-x_0)^2 + (v-y_0)^2 = r^2$$
$$\wedge\ \forall x, y \bullet y = m*x + b\ \wedge\ (x-x_0)^2 + (y-y_0)^2 = r^2\ \Rightarrow\ u^2 + v^2 \le x^2 + y^2\ ]$$

6. For each of the following statements,

- encode it into the language of typed predicate logic and
- translate the result into a formula with untyped quantifiers.

a. *Courses with prerequisites are not taken by first-semester students who have not taken related high-school courses*.

Let

Course be the type for courses
Student be the type for students
prereq(*x*, *y*) mean *x is a prerequisite for y* (intending *x*, *y* : *Course*)
first_semester(*x*) means *x is a first-semester* (*student*)
high_school(*x*) mean *x is a high-school* (*course*)
realated_to(*x*, *y*) mean *x is related to y* (intending *x*, *y* : *Course*)
takes(*x*, *y*) mean *x* (a student) *takes y* (a course)

Answer

$\forall\ x : Course,\ y : Student \bullet (\ \exists\ z : Course \bullet prereq(z, x)\ ) \wedge first\_semester(y)$
$\quad \wedge (\ \forall\ u : Course \bullet high\_school(u) \wedge related\_to(u, x) \Rightarrow \neg\ takes(y, u)\ )$
$\Rightarrow \neg\ takes(y, x)$

$\langle\equiv\rangle$

$\forall\ x : Course,\ y : Student \bullet (\ \exists\ z \bullet course(z) \wedge prereq(z, x)\ ) \wedge first\_semester(y)$
$\quad \wedge (\ \forall\ u \bullet course(\mathrm{u})\ \mathrm{z}\ (\ high\_school(u) \wedge related\_to(u, x) \Rightarrow \neg\ takes(y, u)\ )\ )$
$\Rightarrow \neg\ takes(y, x)$

$\langle\equiv\rangle$

$\forall\ x, y \bullet course(x) \Rightarrow (\ student(y) \Rightarrow (\ (\ \exists\ z \bullet course(z) \wedge prereq(z, x)\ ) \wedge first\_semester(y)$
$\quad \wedge (\ \forall\ u \bullet course(\mathrm{u}) \Rightarrow (\ high\_school(u) \wedge related\_to(u, x) \Rightarrow \neg\ takes(y, u)\ )\ )$
$\Rightarrow \neg\ takes(y, x)\ )\ )$

This is logically equivalent to

$\forall\ x, y \bullet course(x) \wedge student(y) \wedge (\ \exists\ z \bullet course(z) \wedge prereq(z, x)\ ) \wedge first\_semester(y)$
$\quad \wedge (\ \forall\ u \bullet course(\mathrm{u}) \wedge high\_school(u) \wedge related\_to(u, x) \Rightarrow \neg\ takes(y, u)\ )$
$\Rightarrow \neg\ takes(y, x)$

b. *Some buildings have passed the fire inspection yet, if they catch fire, some in them will be in danger unless the fire is quickly extinguished.*

Let

Building be the type for buildings

Inspector be the type for inspectors

Fire be the type for fires

Person be the type for persons

passed_for_fire(x, y) mean x (an inspector) *has passed* y (a building) *in a fire inspection*

burns_in(x, y) mean x (a fire) *burns in* y (a building)

extinguished_quickly(x) mean x (a fire) *is quickly extinguished*

threatens(x, y) mean x (a fire) *threatens* y (a person)

Answer

$\exists\, x : Building \bullet (\, \exists\, y : Inspector \bullet passed\_for\_fire(y, x)\,)$
$\quad \wedge (\, \forall\, z : Fire \bullet burns\_in(z, x)$
$\quad\quad \Rightarrow (\, \neg\, extinguished\_quickly(z) \Rightarrow \exists\, u : Person \bullet threatens(z, u)\,)\,)$

$\langle\equiv\rangle$

$\exists\, x \bullet building(\mathrm{x}) \wedge (\, \exists\, y \bullet inspector(\mathrm{y}) \wedge passed\_for\_fire(y, x)\,)$
$\quad \wedge (\, \forall\, z : Fire \bullet burns\_in(z, x)$
$\quad\quad \Rightarrow (\, \neg\, extinguished\_quickly(z) \Rightarrow \exists\, u \bullet person(u) \wedge threatens(z, u)\,)\,)$

$\langle\equiv\rangle$

$\exists\, x \bullet building(\mathrm{x}) \wedge (\, \exists\, y \bullet inspector(\mathrm{y}) \wedge passed\_for\_fire(y, x)\,)$
$\quad \wedge (\, \forall\, z \bullet fire(z) \Rightarrow (\, burns\_in(z, x)$
$\quad\quad \Rightarrow (\, \neg\, extinguished\_quickly(z) \Rightarrow \exists\, u \bullet person(u) \wedge threatens(z, u)\,)\,)\,)$

This is logically equivalent to

$\exists\, x \bullet building(\mathrm{x}) \wedge (\, \exists\, y \bullet inspector(\mathrm{y}) \wedge passed\_for\_fire(y, x)\,)$
$\quad \wedge (\, \forall\, z \bullet fire(z) \wedge burns\_in(z, x) \wedge \neg\, extinguished\_quickly(z)$
$\quad\quad \Rightarrow \exists\, u \bullet person(u) \wedge threatens(z, u)\,)$

7. Consider the statement

Any person without a permit who parks a commercial truck on a residential street is fined.

a. Encode this into the language of predicate logic with types and constraints. Include a constraint with each quantifier. Let

Person be the type for persons
Truck be the type for trucks
Street be the type for streets
has_permit(x) mean *x has a permit*
residential(x) mean *x is residential*
commercial(x) mean *x is residential*
parks(x, y, z) mean *x parks y on z*
fined(x) mean *x is fined*

Answer

$\forall\ x : Person \mid \neg\ has\_permit(x) \bullet (\ \exists\ y : Truck \mid commercial(y) \bullet$
$\quad \exists\ z : Street \mid residential(z) \bullet parks(x, y, z)\ ) \Rightarrow fined(x)$

Another, logically equivalent way of expressing this is

$\forall\ x : Person \mid \neg\ has\_permit(x) \bullet \forall\ y : Truck \mid commercial(y) \bullet$
$\quad \forall\ z : Street \mid residential(z) \bullet parks(x, y, z) \Rightarrow fined(x)$

b. Translate the result in a into a formula without constraints (but with typed quantifiers).

Answer

$\forall\ x : Person \bullet \neg\ has\_permit(x) \Rightarrow (\ (\ \exists\ y : Truck \bullet commercial(y) \wedge \exists\ z : Street \bullet$
$\quad residential(z) \wedge parks(x, y, z)\ )\ \Rightarrow fined(x)\ )$

This is logically equivalent to

$\forall\ x : Person \bullet \neg\ has\_permit(x) \wedge (\ \exists\ y : Truck \bullet commercial(y) \wedge \exists\ z : Street \bullet$
$\quad residential(z) \wedge parks(x, y, z)\ ) \Rightarrow fined(x)$

c. Translate the result in b into a formula without typed quantifiers.

Answer

$\forall\ x : person(x) \Rightarrow (\ \neg\ has\_permit(x) \wedge (\ \exists\ y \bullet truck(y) \wedge commercial(y) \wedge \exists\ z \bullet$
$\quad street(z) \wedge residential(z) \wedge parks(x, y, z)\ ) \Rightarrow fined(x)\ )$

This is logically equivalent to

$\forall\ x : person(x) \wedge \neg\ has\_permit(x) \wedge (\ \exists\ y \bullet truck(y) \wedge commercial(y) \wedge \exists\ z \bullet$
$\quad street(z) \wedge residential(z) \wedge parks(x, y, z)\ ) \Rightarrow fined(x)$

If we carried the alternative noted in **a** through these steps, we'd end up with

$\forall\ x, y, z \bullet person(x) \wedge \neg\ has\_permit(x) \wedge truck(y) \wedge commercial(y) \wedge street(z)$
$\quad \wedge\ residential(z) \wedge parks(x, y, z) \Rightarrow fined(x)$