Quantificational Logic, part 2

Review of Part 1

```
(x)(Fx \cdot Gx)
                            Valid
                                                                                       Invalid
                                                          (x)(Lx \supset Fx)
 [ : (x)Fx
                                                          (\exists x)Lx
                                                                                         a, b
                                                      [ : (x)Fx
     asm: \sim(x)Fx
                                                                                        La. Fa
                                          * 3 asm: \sim(x)Fx
                                              * 4 \therefore (\existsx)~Fx {from 2}
                                                  4 \therefore (\exists x) \sim Fx \quad \{\text{from 3}\}\
                                                                                     ~Lb, ~Fb
     ∴ ~Fa {from 3}
     \therefore (Fa • Ga) {from 1}
     ∴ Fa {from 5}
                                                        ∴ ~Fb {from 4}
7 : (x)Fx \{ from 2; 4 contradicts 6 \}
                                                         \therefore (La \supset Fa) {from 1}
                                                          \therefore (Lb \supset Fb) {from 1}
                                                          \therefore Fa {from 5 and 7}
                                                         ∴ ~Lb {from 6 and 8}
```

- 1. Reverse squiggles.
- 2. Drop initial existentials, using a new letter each time.
- 3. Lastly, drop initial universals, using all the old letters. (Only use a new letter if you've done everything else possible, including further assumptions if needed, and still have no old letters.)
- 4. If you can't get a contradiction, construct a refutation.

Identity Logic

```
r=l = Romeo is the lover of Juliet. (identity)
```

Ir = Romeo is Italian. (predication)

 $(\exists x)Ix$ = There are Italians. (existence)

The result of writing a small letter and then "=" and then a small letter is a wff.

Romeo isn't the lover of Juliet = $\sim r=1$

Someone besides Romeo is Italian Someone who isn't Romeo is Italian = $(\exists x)(\sim x=r \cdot Ix)$

Romeo is Italian Romeo is Italian but no one else is $(Ir \cdot \sim (\exists x)(\sim x = r \cdot Ix))$

LogiCola H (IM & IT)

There is exactly one Italian = $(\exists x)(Ix \cdot \sim (\exists y)(\sim y = x \cdot Iy))$

There are at least two Italians = $(\exists x)(\exists y)((Ix \cdot Iy) \cdot \sim x = y)$

There are exactly two Italians = $\frac{(\exists x)(\exists y)(((Ix \cdot Iy) \cdot \neg x = y) \cdot \neg x = y)}{\neg (\exists z)((\neg z = x \cdot \neg z = y) \cdot Iz))}$

1 + 1 = 2

If exactly one being is F and exactly one being is G and nothing is F-and-G, then exactly two beings are F-or-G.

```
((((\exists x)(Fx \cdot \sim (\exists y)(\sim y = x \cdot Fy)) \cdot (\exists x)(Gx \cdot \sim (\exists y)(\sim y = x \cdot Gy)))\cdot \sim (\exists x)(Fx \cdot Gx)) \supset(\exists x)(\exists y)(((Fx \vee Gx) \cdot (Fy \vee Gy)) \cdot (\sim x = y)\cdot \sim (\exists z)((\sim z = x \cdot \sim z = y) \cdot (Fz \vee Gz)))))
```

Identity Principles

Self-identity axiom

a=a

Substitute-equals rule

LogiCola IDC

$$Fa, a=b \rightarrow Fb$$

Can also use:
$$a = b, b = c -> a = c$$

 $a = b -> b = a$
Pages 126 to 129

There's more than one being. (pluralism)

:. It's false that there's exactly one being. (monism)

```
* 1 (\exists x)(\exists y) \sim x = y Valid

[\because \sim (\exists x)(y)y = x]

* 2 asm: (\exists x)(y)y = x

* 3 \therefore (\exists y) \sim a = y {from 1}

4 \therefore \sim a = b {from 3}

5 \therefore (y)y = c {from 2}

6 \therefore a = c {from 5}

7 \therefore b = c {from 5}

8 \therefore a = b {from 6 and 7}

9 \therefore \sim (\exists x)(y)y = x {from 2; 4 contradicts 8}
```

```
LogiCola Set I (DV) - Score (level 8) = 0
File Options Tools Help
     1 Jf
     2 s=f
     3 ~Es
       [∴ ~(x)~(Jx · ~Ex)
     4 r asm: (x)~(Jx · ~Ex)
```

12 ∴ ~(x)~(Jx · ~Ex) {from 4; 5 contradicts 11}

6 ∴ Js {from 1 and 2} 7 ∴ f=f {from 2 and 2} 8 ∴ s=s {from 2 and 7} * 9 ∴ ~(Jf • ~Ef) {from 4}

11 L ∴ Ef {from 1 and 9}

10 ∴ ~(Js • ~Es) {from 2 and 9}

Valid This proof shows that

the argument is valid.

LogiCola Set I (DI) - Score (level 8) = 0

File Options Tools Help

1 Jk = 1

* 2
$$\sim (\exists x) \sim (\sim Ex \supset Jx)$$
 = 1

$$2 \sim (\exists x) \sim (\sim Ex \supset Jx) = 1$$
$$[\therefore (\sim Ep \supset k=p) = 0$$

* 3 asm: ~(~Ep ⊃ k=p)

7 ∴ (~Ek ⊃ Jk) {from 4}

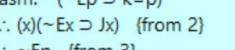
* 8 ∴ (~Ep ⊃ Jp) {from 4}

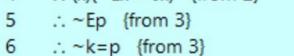
REFUTE

9 ∴ Jp {from 5 and 8}

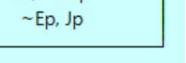
asm:
$$\sim (\sim Ep \supset K=p)$$

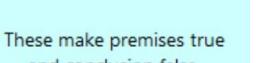
 $\therefore (x)(\sim Ex \supset Jx) \{from 2\}$

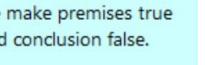


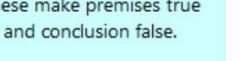












Relational Logic

Lrj = Romeo loves Juliet.

Bxyz = x is between y and z.

The result of writing a capital letter and then two or more small letters is a wff.

L(x) – property of x Can be extended to relation... L(r, j), B(x, y, z) are OK

```
Juliet loves Romeo = Ljr

Juliet loves herself = Ljj

Juliet loves Romeo but not Paris = (Ljr · ~Ljp)

Juliet is between Paris and Romeo = Bjpr
```

```
Everyone loves him/herself = (x)Lxx

Someone loves himself = (\exists x)Lxx

No one loves himself = \sim (\exists x)Lxx
```

Put quantifiers before relations.		
Someone loves Juliet		
For some x, x loves Juliet	For some x, Juliet loves x	
$(\exists x)Lxj$ $(\exists x)Ljx$		

For some, use "." For all, use "⊃"

LogiCola H (RM & RT)

Everyone loves Ju	uliet = $(x)Lxj$	Juliet loves every	one = $(x)Ljx$
For all x,	x loves Juliet	For all x,	Juliet loves x
No one loves Juliet = $\sim (\exists x)Lxj$		Juliet loves no one = $\sim (\exists x) Ljx$	
It is not the case that, for some x,	x loves Juliet	It is not the case that, for some x,	Juliet loves x

Some Montague loves Juliet = $(\exists x)(Mx \cdot Lxj)$

For some x, x is a Montague and x loves Juliet

All Montagues love Juliet = $(x)(Mx \supset Lxj)$

For all x, if x is a Montague then x loves Juliet

Romeo loves some Capulet = $(\exists x)(Cx \cdot Lrx)$

For some x, x is a Capulet and Romeo loves x

Romeo loves all Capulets = $(x)(Cx \supset Lrx)$

For all x, if x is a Capulet then Romeo loves x

All Montagues love themselves = $(x)(Mx \supset Lxx)$		
For all x,	x is a Montague then	x loves x

Some Montague besides Romeo loves Juliet = $(\exists x)((Mx \cdot \sim x=r) \cdot Lxj)$			
	For some x,	x is a Montague and x isn't Romeo and	x loves Juliet

Romeo loves all Capulets who love themselves = $(x)((Cx \cdot Lxx) \supset Lrx)$			
	For all x,	if x is a Capulet and x loves x then	Romeo loves x

These have two relations

All who know Juliet love Juliet = $(x)(Kxj \supset Lxj)$

E 11	·C 1 T 1' / /1	1 T 11
For all x,	if x knows Juliet then	x loves Juliet

All who know themselves love themselves = $(x)(Kxx \supset Lxx)$

For all x,	if x knows x then	x loves x
		THE REPORT OF STREET

These have two quantifiers

Someone loves someone = $(\exists x)(\exists y)Lxy$ For some x and for some y, x loves y Not the same person so two variables

Everyone loves everyone = (x)(y)LxyFor all x and for all y, x loves y

Everyone loves everyone else = $(x)(y)(\sim x=y \supset Lxy)$

For all x and for all y, if x isn't y then x loves y

Some	Montague hates so	me Capulet = $(\exists x)(\exists y)$	$y)((Mx \cdot Cy) \cdot Hxy)$
	For some x and for some y,	x is a Montague and y is a Capulet and	x hates y

Every	y Montague hates	s every Capulet $= (x)(y)$	$((Mx \cdot Cy) \supset Hxy)$
	For all x and	if x is a Montague and	x hates y
	for all y,	y is a Capulet then	X Hates y

Everyone loves someone.

For all x there's some y, such that x loves y.

 $(x)(\exists y)Lxy$

There's someone who everyone loves.

There's some y such that, for all x, x loves y.

 $(\exists y)(x)Lxy$

Variables are just dummy names
Variables should always be quantified
Order and scope of quantifiers are important

LogiCola H (RM & RT)

Pages R4 to R8

Every Capulet loves some Montague

- = (x)(Cx \supset x loves some Montague)
 - $= (x)(Cx \supset (\exists y)(My \cdot Lxy))$

Every Capulet loves someone

- = (x)(Cx \supset x loves someone)
 - = $(x)(Cx \supset (\exists y)Lxy)$

Everyone loves some Montague

- = (x) x loves some Montague
 - = $(x)(\exists y)(My \cdot Lxy)$

```
Some Capulet loves every Montague
= (\exists x)(Cx \cdot x \text{ loves every Montague})
= (\exists x)(Cx \cdot (y)(My \supset Lxy))
```

Some Capulet loves everyone = $(\exists x)(Cx \cdot x \text{ loves everyone})$ = $(\exists x)(Cx \cdot (y)Lxy)$

Someone loves every Montague = $(\exists x)$ x loves every Montague = $(\exists x)(y)(My \supset Lxy)$

```
There is an unloved lover
```

- = $(\exists x)$ (no one loves $x \cdot x$ loves someone)
 - $= (\exists x)(\sim(\exists y)Lyx \cdot (\exists y)Lxy)$

Note y scopes differently; can use y, z

Everyone loves every lover

- = $(x)(x \text{ loves someone } \supseteq \text{ everyone loves } x)$
 - = $(x)((\exists y)Lxy \supset (y)Lyx)$

Romeo loves all and only those who don't love themselves

- = (x)(Romeo loves $x \equiv x$ doesn't love x)
 - = $(x)(Lrx \equiv \sim Lxx)$

All who know any person love that person

- = (x)(y)(x knows y \supset x loves y)
 - = $(x)(y)(Kxy \supset Lxy)$

Reflexive / Irreflexive

Everyone loves himself = (x)LxxNo one loves himself = $(x)\sim Lxx$

Symmetrical / Asymmetrical

```
Universally, if x loves y then = (x)(y)(Lxy \supset Lyx)
y loves x [does not love x] = (x)(y)(Lxy \supset \sim Lyx)
```

Transitive / Intransitive

```
Universally, if x loves y and y loves = (x)(y)(z)((Lxy \cdot Lyz) \supset Lxz)
z, then x loves z [does not love z] = (x)(y)(z)((Lxy \cdot Lyz) \supset \sim Lxz)
```

```
1 (x)Lxx Valid
[ \therefore (x)(\exists y)Lxy 
* 2 asm: \sim (x)(\exists y)Lxy
* 3 \therefore (\exists x) \sim (\exists y)Lxy  {from 2}
* 4 \therefore \sim (\exists y)Lay  {from 3}
5 \therefore (y) \sim Lay  {from 4}
6 \therefore \sim Laa  {from 5}
7 \therefore Laa  {from 1}
8 \therefore (x)(\exists y)Lxy  {from 2; 4 contradicts 6}
```

Relational proofs are often tricky, even though they use no new inference rules. When you have a string of quantifiers, as in lines 2 and 3 above, work on one at a time, starting from the outside. Drop only *initial* quantifiers!

```
Invalid
          (x)Lxx
     [ :: (\exists x)(y)Lyx
                                                        a, b
* 2 asm: \sim (\exists x)(y)Lyx
                                                    Laa, Lbb
    3 \therefore (x) \sim (y) Lyx \quad \{from 2\}
                                                  ~Lba, ~Lab
    4 ∴ Laa {from 1}
* 5 \therefore ~(y)Lya {from 3}
                                                                LogiCola I (RI)
* 6 \therefore (\existsy)~Lya {from 5}
    7 \therefore \sim Lba \{from 6\}
    8 ∴ Lbb {from 1}
* 9 \therefore ~(y)Lyb {from 3}
   10 \therefore (\exists y) \sim Lyb \quad \{\text{from } 9\} \dots \rightarrow \text{get } c, d, \dots
```

If you see an infinite loop coming, break out of it and invent your own refutation.

Endless Loops

Since everyone loves someone a loves someone, call this person b b loves someone, call this person c c loves someone, call this person d

. . .

$$(\exists y) Lay \Rightarrow Lab$$

$$(\exists y) Lby \Rightarrow Lbc$$

$$(\exists y) Lcy \Rightarrow Lcd$$

$$\cdots$$

Alonzo Church's Theorem (1931)

The problem of determining validity in relational logic cannot be reduced to an algorithm (a finite mechanical procedure).