PROOF RULES FOR QUANTIFIERS

constants: a - v   variables: w, x, y, z (with subscripts if needed)

The following rules apply only where the quantifier put in or taken out has the entire rest of the line as its scope (i.e, is the main operator). (So, for example, the ∀Elim rule cannot be applied to '¬∀xP(x)' because its main operator is the '¬'.)

Strictly speaking, this restriction means that a series of quantifiers at the beginning of a sentence must be removed or applied one at a time. However, for a series of quantifiers of the same type (multiple universal quantifiers or multiple existential quantifiers) we will allow ourselves the shortcut of removing or introducing more than one at a time. However, when a statement begins with mixed quantifiers (for example: ∀x∃yL(x,y)), the quantifiers must be taken out one at a time, starting at the left. To put in more than one, add one at a time, always adding at the left end. Remember that one quantifier cannot fall within the scope of another for the same variable.

∀ Elim -- Universal Elimination (examples 1-4)

When you have a universal statement (one in which the main operator is a universal quantifier), you may take out the quantifier and replace the variable it was binding throughout the rest of the line by any constant.

1. Every occurrence of the variable bound by the universal quantifier must be replaced, and all of them must be replaced by the same constant.
2. You may apply ∀ Elim to the same line as many times you choose, replacing the variable with whatever constant you wish on each application of the rule.

∃Intro -- Existential Introduction (examples 5-9)

You may substitute a variable for one or more occurrences of a single constant in a sentence you already have, putting an existential quantifier for that variable at the beginning of the line (with the rest of the line in its scope).

∀ Intro -- Universal Introduction (examples 10-14)

Start a subproof with a boxed constant not found outside that subproof. Replace all occurrences of the constant with the same variable (for either form of ∀ Intro). No other quantifier for the same variable can occur within the scope of the new quantifier.

FOR SIMPLICITY, we will use only the basic Universal Introduction rule, although the text also uses a variation they call General Conditional Proof.

UNIVERSAL INTRODUCTION

```
i |  | [c]
  :  :
n |  | P(c)
  | ∀xP(x)          ∀Intro: i-n
```

∃Elim -- Existential Elimination (examples 15-19)

Apply this rule to a line with an existential sentence. Begin a subproof with a boxed constant that does not occur outside the subproof. To form the first line of the subproof, drop the existential quantifier from the sentence to which you plan to apply this rule; in the rest of the line, replace every occurrence of the variable for that quantifier with the constant introduced in the box. The ∃E rule takes a line from this subproof back out of it (one column to the left). The line that moves to the left cannot contain the (boxed) constant introduced in the first line of the subderivation.

EXAMPLES USING QUANTIFIER RULES

1 and 2 are OK — 3 and 4 are NOT

(1)

1	□x(H(x) → G(x))	P
2	H(a) □ H(b)	P
3	H(a) → G(a)	"E:1
4	H(a)	□E:2
5	G(a)	→E: 3,4
6	H(b) → G(b)	"E:1
7	H(b)	□E: 2
8	G(b)	→ E: 6,7
9	G(a) □ G(b)	□I:5,8

(2)

1	B(a)	P
2	□x(B(x) □ C(x))	P
3	□x(C(x) → D(x))	P
4	B(a) □ C(a)	□E:2
5	C(a)	□ E:1,4
6	C(a) → D(a)	□E:3
7	D(a)	→E: 5,6

[3]

1	□x(S(x) □ N(x))	P
2	S(a)	P
3	S(a) □ N(x)	□E:1
4	N(x)	□ E: 2,3

[4]

1	□x(P(x) → Q(x))	P
2	P(a) □ R(b)	P
3	P(a) → Q(b)	□E:1
4	P(a)	□E:2
5	Q(b)	→E:3,4

5,6, & 7 are OK — 8 and 9 are NOT

(5)

1	N(a) □ R(a)	P
2	□x(N(x) □ R(x))	□I:1

(6)

1	N(a) □ R(a)	P
2	□y(N(a) □ R(y))	□I:1
3	□x□y(N(x) □ R(y))	□I:2

(7)

1	N(a) □ R(b)	P
2	N(a)	1,□E
3	□xN(x)	□I:2
4	R(b)	□E:1
5	□xR(x)	□I:4
6	□xN(x) □□xR(x)	□I:3,5

[8]

1	N(a) □ R(b)	P
2	□x(N(x) □ R(x))	□I:1

[9]

1	□xC(x,a)	P
2	□x□yC(x,y)	□I:1

10 and 11 are OK — 12 and 13 are NOT

(10)

1	□x(J(x) →K(x))	P
2	□x[(J(x) □ K(x)) → L(x)]	P
3	[a]	
4	J(a)	
5	J(a) → K(a)	□E:1
6	K(a)	→E:3,4
7	J(a) □ K(a)	□I:3,5
8	(J(a) □ K(a) →L(a)	□E:2
9	L(a)	→E:6,7
10	J(a) → K(a)	→I: 4-9
11	□x(J(x) → L(x))	□I: 3-10

(11)

1	P(a)	P
2	□xPx → □x[Q(x) □ R(x)]	P
3	□xR(x)	P
4	□xP(x)	□I:1
5	□x(Q(x) □ R(x)]	→E: 2,4
6	[b]	
7	R(b)	□E:3
8	Q(b) □ R(b)	□E:5
9	Q(b)	□ E:7,8
10	□xQx	□I:6-9

[12]

1	□x(G(x) → H(x))	P
2	G(a)	P
3	[a]	
4	G(a) → H(a)	□E:1
5	H(a)	→E:2,4
6	□xH(x)	□I:3-5

[13]

1	□xN(x)	P
2	□xS(x)	P
3	[a]	
4	Na	□E:1
5	[b]	
6	Sb	□E:2
7	Na □ Sb	□I:4,6
8	□x(Nx □ Sx)	□I:5-7

14,15, & 16 are OK — 17 & 18 are NOT

(14)

1	□x(C(x) □ D(x))	P
2	[a] C(a) □ D(a)	
3	D(a)	□E:2
4	□xD(x)	□I:3
5	□xD(x)	□E:1,2-4

(15)

1	□x(R(x) □ □yS(y))	P
2	□xR(x)	P
3	[b] R(b)	
4	R(b) □ □y S(y)	□E:1
5	□yS(y)	□ E:3,4
6	S(a)	□E:5
7	S(a)	□E: 2,3-6

(16)

1	□x[A(x) □ D(x)]	P
2	□xA(x)	P
3	□xD(x) □ □xD(x)	P
4	[b] A(b)	
5	A(b) □ D(b)	□E:1
6	D(b)	□ E:4,5
7	□xD(x)	□I:6
8	□xD(x)	□ E: 3,7
9	□xDx	□E: 2,4-8

[17]

1	□x[L(x) □ M(x)]	P
2	[a] L(a) □ M(a)	
3	M(a)	□E:2
4	M(a)	□E :1,2-3
5	□xM(x)	□I:4

[18]

1	J(a) □ K(a)	P
2	□x[K(x) □ L(x)]	P
3	[a] K(a) □ L(a)	
4	K(a)	□E:3
5	J(a)	□ E:1,4
6	□xJ(x)	□I:5
7	□xJ(x)	□E: 2,3-6

Rules for Derivations with Quantifiers

□ Intro
Using □, join any sentences you already have.

□ Elim
From a conjunction, take any conjunct.

□ Intro
From a sentence and its negation, get □.

□ Elim
From □, get any sentence.

¬ Elim

```
  | …
i | ¬¬A
  | …
  | A      ¬Elim: i
```

¬ Intro

```
  | …
i | | A
  | | …
j | | □
  | …
  | ¬A   ¬Intro: i-j
```

Intro
Using , join any sentence you'd like with any sentence you already have.

Elim

```
  | …
h | A    B
  | …
i | | A
  | | …
j | | C
k | | B
  | | …
l | | C
  | C          Elim: h,i-j,k-l
```

□ Elim
From a conditional and its antecedent, get its consequent.

□ Intro

```
  | …
i | | A
  | | …
j | | B
  | …
  | A □ B          □ Intro: i-j
```

□ Elim
From a biconditional and the sentence on one side of its □ , get the sentence on the other side.

□ Intro

```
  | …
i | | A
  | | …
j | | B
k | | B
  | | …
l | | A
  | A □ B        □ Intro: i-j, k-l
```

□Elim
From a sentence with '□x' as its main operator, drop '□x'; replace all occurrences of 'x' uniformly with any constant.

□ Intro

```
  | …
j | | [a]
  | | …
j | | P
  | □xP(…x…)      □Intro: i-j
```

where 'a' is a constant that does not occur outside the subproof, 'P' is any sentence, 'x' is any variable not occurring in 'P', and 'P(…x…)' results from substituting an 'x' for each occurrence of 'a' in 'P'.

□Intro
In any sentence A you already have, replace 0 or more occurrences of a constant with 'x' (where 'x' is any variable not in A) and apply '□x' to the resulting formula.

□ Elim

```
  | …
i | □xP(…x…)
  | …
j | | [a] P(…a…)
  | | …
k | | Q
  | Q             □Elim: i, j-k
```

where 'a' is a constant that does not occur outside the subproof, 'P(…a…)' results from substituting 'a' for every 'x' in 'P(…x…)', and 'Q' does not contain 'a'.

STRATEGIES FOR PROOFS

1. Try to extract goal statement from a statement you already have, in which the goal statement is a subformula. Instances of a quantified sentence are subformulas of that quantified sentence. The □Elim rule allows you to extract an instance. As before, □Elim, □ Elim, and □ Elim allow you to extract immediate components of formulas. (So if your goal is 'J(a)' and you have ' □x(J(x) □ K(x))', apply □Elim and then □Elim.)

2. If goal sentence cannot be extracted as a whole from any statement you already have, base your strategy on the structure of the goal statement.

□	conjunction	Aim for each conjunct separately, then apply □Intro.
□	conditional	Plan to use □ Intro. To do this, start a subproof with the antecedent as proivisional assumption. Aim for the consequent in the subproof.
□	biconditional	Plan to use □ Intro. Start one subproof with the left side of the biconditional and aim for the right in this subproof. Set up a second subproof going from the right side of the biconditional to the left.
¬	negation	If goal has as its main operator, try reaching it by ¬Intro. To do this, start a subproof with the statement to be negated (but without the ¬) as provisional assumption. Within this subproof, aim for any contradiction you can get.
	disjunction	a) If one disjunct is obviously easy to get, get it, then use Intro. b) If neither disjunct is obviously easy to get, look for an earlier disjunction. Try Elim on earlier disjunction c) If neither of these works, assume the negation of the goal statement. You will then need to use ¬Intro to reach goal.
□	universal	Start a subproof, introducing a constant that does not occur outside the subproof. Aim for an instance of the universal with that constant. Build the universal outside the subproof by □Intro.
□	existential	a) If one instance is obviously easy to get, get it. Then use □Intro. b) If no instance is obviously easy to get, look for an earlier existential statement. Try □Elim on earlier existential. c) If neither of these works, assume the negation of the goal statement. You will then need to use ¬Intro to reach goal.

3. a) If you have an existential sentence already and you can't use it with any of our easy rules, you will probably have to use □Elim. If you will have to use □Elim, set up for it early. (An existential sentence can be used with an easy rule if the existential forms the antecedent of a conditional or one side of a biconditional on another line.)

   b) If you have a disjunction already, and you can't use it with an easy rule, you'll probably have to use Elim rule. If you'll have to use Elim, set up for it early.

4. When aiming for □, look for a negation you already have to use as one member of the contradictory pair of sentences.

5. If you have no idea what to do, try applying any easy rules you can. Perhaps the results of this process will give you some ideas for other things to do.

6. When all else fails, assume the opposite of what you want.

EXAMPLES USING THE ∃Elim RULE

	Formula	Rule	Commentary
1	$\exists x(S(x) \land L(x))$		At least one (perhaps former) Senator has lied under oath..
2	$\forall x(S(x) \rightarrow P(x))$		Every Senator is a politician.
3	| [a] $S(a) \land L(a)$		Let's suppose *a* is a Senator who has lied under oath. (Line 1 says there's at least one, but doesn't identify one. We are temporarily assuming a is such a Senator.)
4	| $S(a)$	∧ Elim: 3	On this supposition, *a* is a Senator.
5	| $S(a) \rightarrow P(a)$	∀ Elim: 2	If *a* is a Senator, *a* is a politician.
6	| $P(a)$	→ Elim: 4,5	So *a* is a politician.
7	| $L(a)$	∧ Elim: 3	Also, *a* has lied under oath.
8	| $P(a) \land L(a)$	∧Intro: 6,7	So *a* both is a politician and has lied under oath.
9	| $\exists x(P(x) \land L(x))$	∃ Intro: 8	ON OUR ASSUMPTION, at least one politician has lied under oath.
10	$\exists x(P(x) \land L(x))$	∃Elim: 1,3-9	So at least one politician has lied under oath. (This conclusion doesn't depend on the truth of our assumption about *a*, because we could have reached the same conclusion no matter what constant had been used in step 3.)

	Formula	Rule	Commentary
1	$\forall x(P(x) \rightarrow \neg H(x))$		All politicians are dishonest.
2	| $\exists x(P(x) \land H(x))$		Suppose there is a politician who is honest.
3	| | [a] $P(a) \land H(a)$		Let's assume for now *a* is such a person.
4	| | $P(a) \rightarrow \neg H(a)$	∀Elim: 1	THEN, if *a* is a politician, *a* is dishonest.
5	| | $P(a)$	∧Elim: 3	Also, *a* is a politician,
6	| | $\neg H(a)$	→ Elim:4,5	so *a* is dishonest.
7	| | $H(a)$	∧Elim: 3	But (we were already told) *a* is honest.
8	| | ⊥	⊥ Intro: 6,7	Contradiction: *a* both is and is not honest.
9	| ⊥	∃ Elim: 2,3-8	The contradiction follows from the claim that there's an honest politician, regardless of who might be one. (It doesn't depend on *a* in particular being such a person. Anyone we might pick as an honest politician could be shown, as *a* was, both to be honest and not to be honest.)
10	$\neg\exists x(P(x) \land H(x))$	¬Intro: 2-9	So it's not true that there's an honest politician (because the assumption that there is one leads to a contradiction). In other words, no politicians are honest.

	Formula	Rule
1	$\forall x(P(x) \rightarrow Q(x,a))$	
2	| [b] $\exists x(P(x) \land \neg Q(x,a))$	
3	| | $P(b) \land \neg Q(b,a)$	
4	| | $P(b) \rightarrow Q(b,a)$	∀Elim: 1
5	| | $P(b)$	∧ Elim: 3
6	| | $Q(b,a)$	→ Elim: 4,5
7	| | $\neg Q(b,a)$	∧ Elim: 3
8	| | ⊥	⊥ Intro: 6,7
9	| ⊥	∃ Elim: 2,3-8
10	$\neg\exists x(P(x) \land \neg Q(x,a))$	¬ Intro: 2-9

EXTRA PROBLEMS FOR CHAPTER 13, GROUP 1

Start by opening Proof CStern 130x or Sentences CStern 130x for H 13.x,

If the argument is FO-valid, use Fitch to give a proof. Use AnaCon only for literals (atomic sentences and their negations). Do not use TautCon.
If the argument is FO-invalid, use Tarski's World to give a counterexample. Save worlds as World CStern 130x or 13x, corresponding to Sentences CStern 130x or 13x.

H 13.1
□x (Small(x) → Cube(x))
□x Dodec(x)
□x¬Small(x)

H 13.2
□x((Small(x)   Large(x)) → Tet(x))
□x (Small(x)   Medium(x)   Large(x))
□x Cube(x)
□x Medium(x)

H 13.3
□x Dodec(x) → □x Small(x)
□x (Dodec(x) → Small(x))

H 13.4
□x (Large(x,a) □ Cube(x))
□x Large(x,a) □ □x Cube(x)

H 13.5
□x Larger(x,a)
□x Cube(x)
□x (Larger(x,a) □ Cube(x))

H 13.6
□x (Dodec(x) → Medium(x))
□x Large(x) → □x ¬Dodec(x)

H 13.7
□x(Small(x)   Large(x))
□x(Small(x) □   Tet(x))
□x(Dodec(x)   ¬Large(x))
□x(Tet(x)   Dodec(x))

H 13.8
□x(Dodec(x) → Large(x))
¬□x(Small(x) □ Cube(x))
□x((Cube(x)   Dodec(x)) → ¬Small(x))