Announcements & Such

- *Grand Funk Railroad*
- Administrative Stuff
  - My take-home mid-term solutions have been posted.
    * These are worth studying. Some interesting things there.
  - We will be discussing the grade curve for the course as soon as all of the mid-term grades are in (I will do this on Thursday).
  - **HW #4 is due Thursday (first submission).**
- Today: Chapter 4 — Natural Deduction Proofs for LSL
  - Today: *more proofs* using the basic natural-deduction rules.
  - Plus, a couple more topics from chapter 4 (↔ rules, and SI/TI).
  - Then, it's on to Chapter 5 — (Monadic) Predicate Logic!
  - **MacLogic** — a useful computer program for natural deduction.
    * See http://fitelson.org/maclogic.htm.

☞ **Make sure you do lots of proofs — practice is the key here.**

10 More Examples Involving ∨I and ∨E

1. $(A \,\&\, B) \vee (A \,\&\, C) \vdash A$ [p. 111, ex. 2]
2. $(A \rightarrow \curlywedge) \vee (B \rightarrow \curlywedge), B \vdash \sim A$ [p. 116, §4.5, ex. 11]
3. $(A \vee B) \vee C \vdash A \vee (B \vee C)$ [p. 116, ex. 19]
4. $A \vee B \vdash (A \rightarrow B) \rightarrow B$ [p. 116, ex. 10]
5. $A \,\&\, B \vdash \sim(\sim A \vee \sim B)$ [p. 116, ex. 14 ($\vdash$)]
6. $A \vee B \vdash \sim(\sim A \,\&\, \sim B)$ [p. 116, ex. 13]
7. $\sim(A \,\&\, B) \vdash \sim A \vee \sim B$ [p. 116, ex. 16 ($\dashv$)]
8. $\sim C \vee (A \rightarrow B) \vdash (C \,\&\, A) \rightarrow B$ [not in text]
9. $\vdash (A \rightarrow B) \vee (B \rightarrow A)$ [not in text]
10. $\sim(A \vee B) \vdash \sim A \,\&\, \sim B$ [not in text]

Proof of Example #7

Problem is: ~(A&B) ⊢ ~A∨~B

1	(1)	~(A&B)	Premise
2	(2)	~(~A∨~B)	Assumption (~I)
3	(3)	~A	Assumption (~I)
3	(4)	~A∨~B	3 ∨I
2,3	(5)	⋏	2,4 ~E
2	(6)	~~A	3,5 ~I
2	(7)	A	6 DN
8	(8)	~B	Assumption (~I)
8	(9)	~A∨~B	8 ∨I
2,8	(10)	⋏	2,9 ~E
2	(11)	~~B	8,10 ~I
2	(12)	B	11 DN
2	(13)	A&B	7,12 &I
1,2	(14)	⋏	1,13 ~E
1	(15)	~~(~A∨~B)	2,14 ~I
1	(16)	~A∨~B	15 DN

Proof of Example #8

Problem is: ~C∨(A→B) ⊢ (C&A)→B

1	(1)	~C∨(A→B)	Premise
2	(2)	C&A	Assumption (→I)
3	(3)	~B	Assumption (~I)
4	(4)	~C	Assumption (∨E)
2	(5)	C	2 &E
2,4	(6)	⋏	4,5 ~E
7	(7)	A→B	Assumption (∨E)
2	(8)	A	2 &E
2,7	(9)	B	7,8 →E
2,3,7	(10)	⋏	3,9 ~E
1,2,3	(11)	⋏	1,4,6,7,10 ∨E
1,2	(12)	~~B	3,11 ~I
1,2	(13)	B	12 DN
1	(14)	(C&A)→B	2,13 →I

Proof of Example #9

Problem is: ⊢ (A → B) ∨ (B → A)

1	(1)	~((A→B)∨(B→A))	Assumption (~I)
2	(2)	B	Assumption (→I)
3	(3)	~A	Assumption (~I)
4	(4)	A	Assumption (→I)
2	(5)	A→B	4,2 →I
2	(6)	(A→B)∨(B→A)	5 ∨I
1,2	(7)	Λ	1,6 ~E
1,2	(8)	~~A	3,7 ~I
1,2	(9)	A	8 DN
1	(10)	B→A	2,9 →I
1	(11)	(A→B)∨(B→A)	10 ∨I
1	(12)	Λ	1,11 ~E
	(13)	~~((A→B)∨(B→A))	1,12 ~I
	(14)	(A→B)∨(B→A)	13 DN

Proof of Example #10

Problem is : ~(A∨B) ⊢ ~A&~B

1	(1)	~(A∨B)	Premise
2	(2)	A	Ass (~I)
2	(3)	A∨B	2 ∨I
1,2	(4)	Λ	1,3 ~E
1	(5)	~A	2,4 ~I
6	(6)	B	Ass (~I)
6	(7)	A∨B	6 ∨I
1,6	(8)	Λ	1,7 ~E
1	(9)	~B	6,8 ~I
1	(10)	~A&~B	5,9 &I

The Rule of Definition for the Biconditional

Rule of Definition for ↔ (Df): If ⌜$(p \to q) \,\&\, (q \to p)$⌝ occurs as the entire formula at line j, then at line k we may write ⌜$p \leftrightarrow q$⌝, labeling the line 'j Df' and writing on its left the same numbers as are on the left of j. Conversely, if ⌜$p \leftrightarrow q$⌝ occurs as the entire formula at a line j, then at line k we may write ⌜$(p \to q) \,\&\, (q \to p)$⌝, labeling the line 'j Df' and writing on its left the same numbers as are on the left of j.

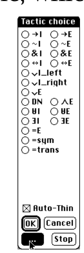
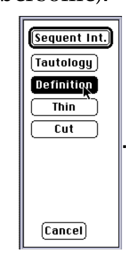
$a_1, \ldots, a_n$	(j)	$(p \to q) \,\&\, (q \to p)$	
	⋮		
$a_1, \ldots, a_n$	(k)	$p \leftrightarrow q$	j Df

OR

$a_1, \ldots, a_n$	(j)	$p \leftrightarrow q$	
	⋮		
$a_1, \ldots, a_n$	(k)	$(p \to q) \,\&\, (q \to p)$	j Df

Using ↔ in MacLogic

- Using the Definition strategy of MacLogic (accessed *via* the [···] button), we can implement our Df. rule for ↔. *Do not use* ↔I *or* ↔E!
- Using MacLogic's Definition strategy is much simpler than using its Tautology strategy (I did that last time, which was cumbersome).

☞ To get to Definition, first:  then .

- Here is a non-trivial example: $A \leftrightarrow \sim B \vdash \sim(A \leftrightarrow B)$. Let's try to tackle this one, using MacLogic's Definition strategy for our Df.
- The shortest proof I've been able to find is 18 steps (next slide). Forbes gives a 20-stepper in his discussion of this example (p. 118).

Problem is : $A \leftrightarrow \sim B \vdash \sim(A \leftrightarrow B)$

1	(1)	$A \leftrightarrow \sim B$	Ass
2	(2)	$A \leftrightarrow B$	Ass
1	(3)	$(A \rightarrow \sim B) \& (\sim B \rightarrow A)$	1 Defn.
1	(4)	$A \rightarrow \sim B$	3 &E
1	(5)	$\sim B \rightarrow A$	3 &E
6	(6)	B	Ass
2	(7)	$(A \rightarrow B) \& (B \rightarrow A)$	2 Defn.
2	(8)	$B \rightarrow A$	7 &E
2,6	(9)	A	8,6 →E
1,2,6	(10)	$\sim B$	4,9 →E
1,2,6	(11)	$\curlywedge$	10,6 ~E
1,2	(12)	$\sim B$	6,11 ~I
1,2	(13)	A	5,12 →E
1,2	(14)	$\sim B$	4,13 →E
2	(15)	$A \rightarrow B$	7 &E
1,2	(16)	B	15,13 →E
1,2	(17)	$\curlywedge$	14,16 ~E
1	(18)	$\sim(A \leftrightarrow B)$	2,17 ~I

Sequent and Theorem Introduction: I

- You may have noticed that certain important sequents or theorems tend to get proven over and over again in different problems.
- For instance, the sequent $X \vee Y, \sim X \vdash Y$ is a very useful thing to know, as are the sequents $X \rightarrow Y, \sim Y \vdash \sim X$, $\curlywedge \vdash X$, and many others.
- It would be nice if we had a rule that allowed us to say "OK, I've proven this sequent already, so I don't have to prove it again here".
- We have two such rules. They are called *Sequent Introduction* (SI) for sequents, and *Theorem Introduction* (TI) for theorems.
- SI and TI allow us to avoid having to re-solve certain sub-problems that we already know how to solve. This makes proofs shorter.
- We will have a fixed list of sequents and theorems that we'll be allowed to use in conjunction with SI and TI.

Sequent and Theorem Introduction: II

- Forbes lists a bunch of sequents and Theorems on page 123 that we may use with SI or TI. There's a MacLogic file containing all of them.
- Here are a few of the sequents and theorems that tend to be useful:

$$p \vee q, \sim p \vdash q; \text{ or; } p \vee q, \sim q \vdash p \qquad \text{(DS)}$$

$$p \rightarrow q, \sim q \vdash \sim p \qquad \text{(MT)}$$

$$p \vdash q \rightarrow p; \text{ or; } \sim p \vdash p \rightarrow q \qquad \text{(PMI)}$$

$$\vdash p \vee \sim p \qquad \text{(LEM)}$$

$$\sim(p \& q) \dashv\vdash \sim p \vee \sim q \qquad \text{(DEM)}$$

$$\sim(p \vee q) \dashv\vdash \sim p \& \sim q \qquad \text{(DEM)}$$

$$\sim(\sim p \vee \sim q) \dashv\vdash p \& q \qquad \text{(DEM)}$$

$$\sim(\sim p \& \sim q) \dashv\vdash p \vee q \qquad \text{(DEM)}$$

$$\curlywedge \vdash p \qquad \text{(EFQ)}$$

$$p \& (q \vee r) \dashv\vdash (p \& q) \vee (p \& r) \qquad \text{(DIST)}$$

Sequent and Theorem Introduction: III

- Remember the proof for #9 above: $\vdash (A \rightarrow B) \vee (B \rightarrow A)$.

1	(1)	$\sim((A \rightarrow B) \vee (B \rightarrow A))$	Assumption (~I)
2	(2)	B	Assumption (→I)
3	(3)	$\sim A$	Assumption (~I)
4	(4)	A	Assumption (→I)
2	(5)	$A \rightarrow B$	4,2 →I
2	(6)	$(A \rightarrow B) \vee (B \rightarrow A)$	5 ∨I
1,2	(7)	$\curlywedge$	1,6 ~E
1,2	(8)	$\sim\sim A$	3,7 ~I
1,2	(9)	A	8 DN
1	(10)	$B \rightarrow A$	2,9 →I
1	(11)	$(A \rightarrow B) \vee (B \rightarrow A)$	10 ∨I
1	(12)	$\curlywedge$	1,11 ~E
	(13)	$\sim\sim((A \rightarrow B) \vee (B \rightarrow A))$	1,12 ~I
	(14)	$(A \rightarrow B) \vee (B \rightarrow A)$	13 DN

Sequent and Theorem Introduction: IV

- Using TI and SI, we can obtain the following much simpler proof:

	(1)	A∨~A	TI (LEM)
2	(2)	A	Assumption (∨E)
2	(3)	B→A	2 SI (PMI)
2	(4)	(A→B)∨(B→A)	3 ∨I
5	(5)	~A	Assumption (∨E)
5	(6)	A→B	5 SI (PMI)
5	(7)	(A→B)∨(B→A)	6 ∨I
	(8)	(A→B)∨(B→A)	1,2,4,5,7 ∨E

- Here, LEM is the theorem $\vdash A \vee \sim A$ (which we have already proven), and PMI stands for either of the sequents $\sim A \vdash A \to B$ (used at line 6), or $A \vdash B \to A$ (used at line 3), both of which we've proven.
- SI allows you to use (*any* substitution instance of) *any* sequent that you've already proven to make an inference at any stage of a proof.
- TI allows you to write down (*any* substitution instance of) *any* theorem that you have already proven at *any* stage of a proof.

The Formal Definitions of SI and TI

- **Sequent Introduction** (SI). Suppose $r_1, \ldots, r_n \vdash s$ is a *substitution-instance* of the sequent $p_1, \ldots, p_n \vdash q$ which we have already proved, and that the formulae $r_1, \ldots, r_n$ occur at lines $j_1, \ldots, j_n$ in a proof. Then we may infer s at line k, labeling the line '$j_1, \ldots, j_n$ SI (Identifier)' and writing on the left all numbers which appear on the left of lines $j_1, \ldots, j_n$.
- **Theorem Introduction** (TI). If $\vdash s$ is a *substitution-instance* of some theorem $\vdash q$ which we have already proved, we may introduce a new line k into a proof with the formula s at it and no numbers on its left, labeling the line 'TI (Identifier)'.
- 'Identifier' stands for the name of a sequent or theorem that has already been proven (*e.g.*, MT, DS, PMI, LEM, *etc*). See Forbes's list.
- Note: TI is just a *special case* of SI (with $n = 0$).

SI and TI: A Relatively Easy Example

- Use SI/TI to find a "short" proof of: $\sim(A \to (B \vee C)) \vdash (B \vee C) \to A$.

Problem is : ~(A→(B∨C)) ⊦ (B∨C)→A

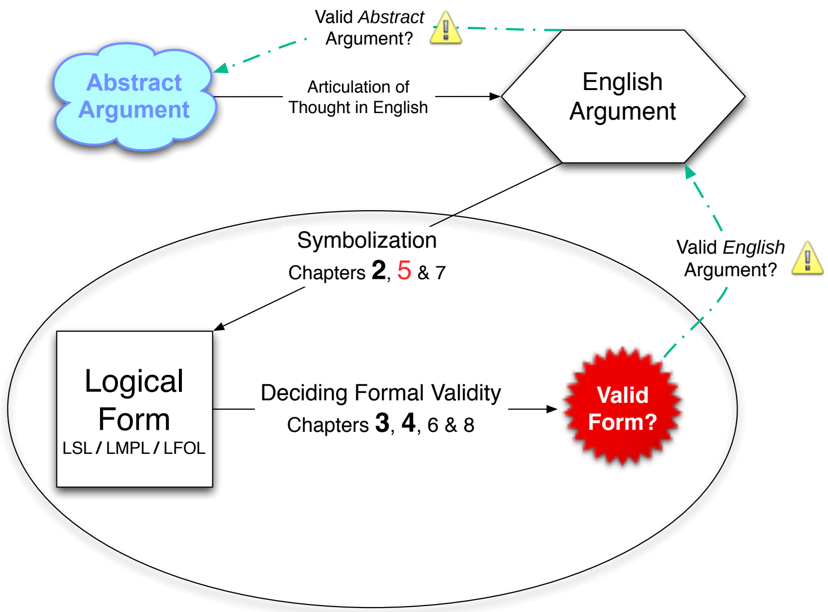
1	(1)	~(A→(B∨C))	Premise
1	(2)	A&~(B∨C)	1 SI Neg-Imp1
1	(3)	A	2 &E
1	(4)	(B∨C)→A	3 SI PMI1

SI and TI: A More Challenging Example

- Use SI/TI to find a "short" proof of: $A \to (B \vee C) \vdash (A \to B) \vee (A \to C)$.

Problem is : A→(B∨C) ⊦ (A→B)∨(A→C)

1	(1)	A→(B∨C)	Premise
1	(2)	~A∨(B∨C)	1 SI IMP1
3	(3)	~A	Assumption (∨E)
3	(4)	A→B	3 SI PMI2
3	(5)	(A→B)∨(A→C)	4 ∨I_left
6	(6)	B∨C	Assumption (∨E)
7	(7)	B	Assumption (∨E)
7	(8)	A→B	7 SI PMI1
7	(9)	(A→B)∨(A→C)	8 ∨I_left
10	(10)	C	Assumption (∨E)
10	(11)	A→C	10 SI PMI1
10	(12)	(A→B)∨(A→C)	11 ∨I_right
6	(13)	(A→B)∨(A→C)	6,7,9,10,12 ∨E
1	(14)	(A→B)∨(A→C)	2,3,5,6,13 ∨E



Chapter 5: Predication and Quantification

- Consider the following two arguments:

① Socrates is wise. ∴ Someone is wise.

② Everyone is happy. ∴ Plato is happy.

- Intuitively, both ① and ② are *valid* (*why*?). But, if we try to translate these into LSL, we get the *in*valid LSL forms:

$①_{\text{LSL}}$ S $\therefore W$

$②_{\text{LSL}}$ H $\therefore P$

- In LSL, we are not able to capture the *logical structure* shared between premises and conclusions of these kinds of arguments.
- If it's not *atomic sentences* that the premises and conclusions of such arguments have in common (structurally), then what *is* it?
- This is what Chapter 5 is about…

Predication and Quantification: II

- We need a *richer language* than LSL — one which accurately captures the deeper *logical structure* of arguments like ① and ②. New Jargon:
- A **predicate** is something which *applies to* an object or *is true of* an object or which an object *satisfies*. *E.g.*, Socrates satisfies the predicate **(is) Wise**.
- A **proper name** is a word or a phrase which *stands for*, or *refers to*, or *denotes* a specific person, place, or thing. *E.g.*, 'Socrates' is a proper name.
- **Quantifier phrases** specify *quantities*. *E.g.*, 'someone' means *at least one* person and 'everyone' means *all* people. 'Some' and 'all' are **quantifiers**.
- The collection of objects to which the quantifiers in a statement are *relativized* is called the **domain of discourse** of the statement (*e.g.*, 'some*one*' quantifies only over *people*, 'some*time*' quantifies over *times*).
- Chapter 5 introduces the logical language LMPL (the Language of Monadic Predicate Logic) that contains these elements (and a few more tricks).

Symbolization in LMPL I: New Atomic Sentences

- Among the atomic sentences of LMPL (*in addition to LSL sentence letters*) are (new) strings of the form ⌜Xn⌝, where 'X' is a (monadic) predicate, and 'n' is an individual constant (*i.e.*, a proper name).
- We will use the lower-case letters 'a'–'s' as *individual constants* ('t'–'z' are used as *variables* — much more on variables later).
- Some examples of these new kinds of atomic sentences:
  - 'Branden is tall.' ↦ 'Tb'.
  - 'Honda is an automobile manufacturer.' ↦ 'Ah'.
  - 'New York is a city.' ↦ 'Cn'.
- As in LSL, we can *combine* different LMPL atomic sentences using the sentential connectives to yield complex sentences. For instance:
  - 'Branden is tall, but Ruth is not tall.' ↦ '$Tb \,\&\, {\sim}Tr$'.