Announcements & Overview

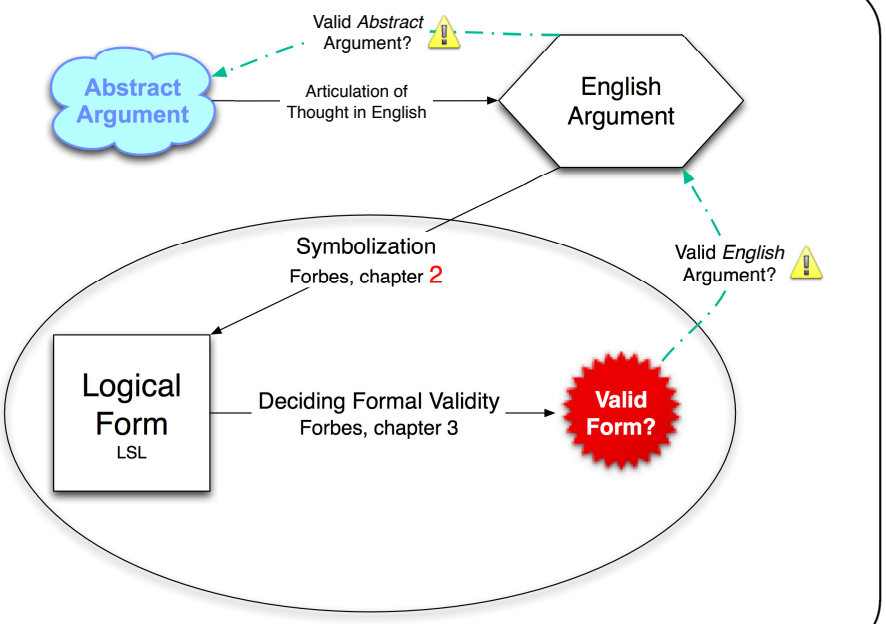
- Administrative Stuff
  - ☞ **HW #1 is due on Friday (*via* `Blackboard`).**
    - Please make sure you're signed-up for TurningPoint Cloud

      `http://bit.ly/TurningPointCloud`
    - We'll use it for some in-class quizzes/surveys (*e.g.*, **today**)
- Today: Introduction to Unit #2 (**L**anguage of **S**entential **L**ogic)
  - But, first, a brief True/False quiz (similar to HW #1 questions)
  - Then, "The Big Picture" View of Part I of the Course
  - Philosophical Preamble to Unit #2 — **The Use/Mention Distinction**
  - Introduction to Unit #2 (LSL)
    - ∗ The five sentential connectives (logical constants) of LSL
    - ∗ As we'll see in Unit #3, these connectives will be ***truth-functional***
    - ∗ First Steps in Symbolization: English ↦ LSL

Ten True/False Questions About Unit #1

1. If the conclusion of an argument cannot (on pain of logical contradiction) be false, then said argument is valid.
2. Joe Biden is currently the President of the United States.
3. An argument with premises that are all actually true and a conclusion that is actually false must be invalid.
4. Some invalid arguments have true conclusions.
5. Some sound arguments have false conclusions.
6. Adding premises to a valid argument can never render it invalid.
7. If an argument is sententially invalid, then it is (absolutely) invalid.
8. The following is an invalid sentential form:

   Either P or Q.
   $\therefore P$.
9. If an argument is an instance of an invalid form, then it is invalid.
10. If an argument is an instance of a valid form, then it is valid.



Preamble for Unit #2: The Use/Mention Distinction

- Consider the following two sentences:

  (1) California has more than nine residents.

  (2) 'California' has more than nine letters.
- In (1), we are *using* the word 'California' to talk about the State of California. But, in (2), we are merely *mentioning* the word 'California' (*i.e.*, we're talking about *the word itself*).
- If Jeremiah = 'California', which of these sentences are true?

  (3) Jeremiah has (exactly) eight letters [false].

  (4) Jeremiah has (exactly) ten letters [true].

  (5) 'Jeremiah' has eight letters [true].

  (6) 'Jeremiah' is the name of a state [false].

Preamble for Unit #2: Further Examples of Use/Mention

- Here are some True/False questions regarding Use/Mention.

(1) $2 + 2 = 4$.

(2) '$2 + 2$' $= 4$.

(3) In sentence (1), no numerals are mentioned (*i.e.*, all are *used*).

(4) Sentence (2) makes a claim about the number two.

(5) Sentence (1) makes a claim about the number four.

(6) The number of sumbols in sentence (1) is '6'.

(7) The number of symbols in sentence (2) is seven.

(8) 'Sentence (8)' is true.

(9) Sentence (6) makes a claim about the Arabic numeral which (when *used*) denotes the number six.

(10) 'The sky is blue.'

Preamble for Unit #2: More on Use/Mention and ' ' *versus* ⌜ ⌝

- Consider the following two statements about LSL sentences

(*i*) If p and q are both sentences of LSL, then so is ⌜$(p \,\&\, q)$⌝.

(*ii*) If p and q are both sentences of LSL, then so is '$(p \,\&\, q)$'.

- As it turns out, (*i*) is true, but (*ii*) is *false*. The string of symbols '$(p \,\&\, q)$' *cannot* be a sentence of LSL, since 'p' and 'q' are *not* part of the lexicon of LSL. They allow us to talk about LSL *forms*.
- The trick is that ⌜$(p \,\&\, q)$⌝ abbreviates the long-winded phrase:
  - The symbol-string which results from writing '(' followed by p followed by '&' followed by q followed by ')'.
- In (*ii*), we are merely *mentioning* 'p' and 'q' (in '$(p \,\&\, q)$'). But, in (*i*), we are *using* 'p' and 'q' (in ⌜$(p \,\&\, q)$⌝) to talk about (forms of) sentences in LSL. In (*i*), 'p' and 'q' are *used* as *metavariables*.

Preamble for Unit #2: Object language, Metalanguage, *etc.* ...

- LSL is the *object language* of our current studies. The symbol string '$(A \,\&\, B) \vee C$' is a sentence of LSL. But, the symbol string '$(p \,\&\, q) \vee r$' is *not* a sentence of LSL. Why?
- We use a *metalanguage* to talk about the object language LSL. This metalanguage is not formalized. It's mainly English, plus *metavariables* like 'p', 'q', 'r', and *selective quotes* '⌜' and '⌝'.
- If $p =$ '$(A \vee B)$', and $q =$ '$(C \to D)$', then what are the following?
  - ⌜$p \,\&\, q$⌝ $[(A \vee B) \,\&\, (C \to D)]$, '$p \,\&\, q$' $[p \,\&\, q]$, 'p' $[p]$, 'q' $[q]$
- And, which of the following are true?
  - p has five symbols [true]. 'p' has five symbols [false].
  - ⌜$p \,\&\, q$⌝ is a sentence of LSL [true]. So is '⌜$p \,\&\, q$⌝' [false].

Introduction to the Syntax of the LSL: The Lexicon

- The syntax of LSL is quite simple. Its lexicon has the following symbols:
  - Upper-case letters 'A', 'B', ... which stand for *basic sentences*.
  - Five *sentential connectives/operators* (one *unary*, four *binary*):

Operator	Name	Logical Function	Used to translate
'$\sim$'	tilde	negation	not, it is not the case that
'&'	ampersand	conjunction	and, also, moreover, but
'$\vee$'	vee	disjunction	or, either ... or ...
'$\to$'	arrow	conditional	if ... then ..., only if
'$\leftrightarrow$'	double arrow	biconditional	if and only if

  - Parentheses '(', ')', brackets '['. ']', and braces '{', '}' for grouping.
- If a string of symbols contains anything else, then it's not a sentence of LSL. And, only *certain* strings of these symbols are LSL sentences.
- Some LSL symbol strings aren't *well-formed*: '$(A \,\&\, B$', '$A \,\&\, B \vee C$', *etc.*

The Five Kinds (Forms) of *Non-Basic* LSL Sentences

- Sentences of the form ⌜$p \,\&\, q$⌝ are called *conjunctions*, and their constituents (p, q) are called *conjuncts*.
- Sentences of the form ⌜$p \vee q$⌝ are called *disjunctions*, and their constituents (p, q) are called *disjuncts*.
- Sentences of the form ⌜$p \to q$⌝ are called *conditionals*. p is called the *antecedent* of ⌜$p \to q$⌝, and q is called its *consequent*.
- Sentences of the form ⌜$p \leftrightarrow q$⌝ are called *biconditionals*. p is called the *left-hand side* of ⌜$p \leftrightarrow q$⌝, and q is its *right-hand side*.
- Sentences of the form ⌜$\sim p$⌝ are called *negations*. The sentence p is called the *negated sentence*.
- These 5 kinds of sentences (+ *atoms*) are the *only* kinds in LSL.
- Next, we begin to think about "translation" from English into LSL.

English ↦ LSL I: Basic Steps Toward Symbolization

- Sentences with *no* connectives are *trivial* to "translate" or symbolize:
  - 'It is cold.' ↦ 'C'.
  - 'It is rainy.' ↦ 'R'.
  - 'It is sunny.' ↦ 'S'.
- Sentences with just one sentential connective are also pretty easy:
  - 'It is cold and rainy.' ↦ '$C \,\&\, R$'. [why two atomic letters?]

☞ Try to give the most *precise* (fine-grained) LSL rendition you can, and try to come as close as possible to capturing the meaning of the original.

- Sentences with two connectives can be trickier:
  - 'Either it is sunny or it is cold and rainy.' ↦ '$S \vee (C \,\&\, R)$'.
- Q: Why is '$(S \vee C) \,\&\, R$' incorrect? A: The English is *not* a conjunction.

English ↦ LSL II: Symbolizing in Two Stages

☞ When symbolizing English sentences in LSL (especially complex ones), it is useful to perform the symbolization in (at least) *two stages*.

Stage 1: Replace all basic sentences (explicit or implicit) with atomic letters. This yields a sentence in "Logish" (neither English nor LSL).

Stage 2: Eliminate remaining English by replacing English connectives with LSL connectives, and properly grouping the resulting symbolic expression (w/parens, *etc.*) to yield pure LSL.

- Here are some simple examples involving only single connectives:

English:	**"Logish":**	**LSL:**
Either it's raining or it's snowing.	Either R or S.	$R \vee S$
If Dell introduces a new line, then Apple will also.	If D, then A.	$D \to A$
Snow is white and the sky is blue.	W and B.	$W \,\&\, B$
It is not the case that Emily Bronte wrote *Jane Eyre*.	It is not the case that E.	$\sim E$
John is a bachelor if and only if he is unmarried.	J if and only if not M.	$J \leftrightarrow \sim M$

English ↦ LSL III: Symbolizations involving '&' and '∨'

- We use '&' to symbolize a variety of English connectives, including:
  - 'and', 'yet', 'but', 'however', 'moreover', 'nevertheless', 'still', 'also', 'although', 'both', 'additionally', 'furthermore' (and others)
- There is often more to the meaning of 'but', 'nevertheless', 'still', 'although', 'however' (and other such English connectives) than merely 'and'. But, in LSL, the closest we can get to these connectives is '&'.
- On the other hand, there are fewer English expressions that we will symbolize using '∨'. Typically, these involve either 'or' or 'either . . . or'.
- But, less typically and more controversially, there is one other English connective we will symbolize as '∨', and that is 'unless'. Seem strange?
- Intuitively, ⌜p unless q⌝ means something like ⌜if not q, then p⌝. But, in LSL, ⌜$\sim q \to p$⌝ is *equivalent* to (*means the same as*) ⌜$p \vee q$⌝. [Ch. 3.]

English ↦ LSL IV: Symbolizations involving '→' (and '↔')

☞ We will use '→' to symbolize *many* different English expressions. These will be the most controversial and tricky of our LSL symbolizations. *E.g.*:

- ⌜if p then q⌝ ↦ ⌜$p \to q$⌝
- ⌜p implies q⌝ ↦ ⌜$p \to q$⌝
- ⌜p only if q⌝ ↦ ⌜$p \to q$⌝
- ⌜q if p⌝ ↦ ⌜$p \to q$⌝
- ⌜p is a sufficient condition for q⌝ ↦ ⌜$p \to q$⌝
- ⌜q is a necessary condition for p⌝ ↦ ⌜$p \to q$⌝
- ⌜q provided p⌝ ↦ ⌜$p \to q$⌝
- ⌜q whenever p⌝ ↦ ⌜$p \to q$⌝
- ⌜p is contingent upon q⌝ ↦ ⌜$p \to q$⌝

- ⌜$p \leftrightarrow q$⌝ is equivalent to ⌜$(p \to q)\,\&\,(q \to p)$⌝ (so mastering '→' is key)

English ↦ LSL V: Grouping Two or More Binary Connectives

- Whenever three or more LSL sentence letters appear in an LSL sentence, parentheses (or brackets or braces) must be used (carefully!) to indicate the intended *scope* of the connectives. Otherwise, problems ensue …
- *E.g.*, '$A\,\&\,B \vee C$' is *not* an LSL sentence. It is *ambiguous* between '$(A\,\&\,B) \vee C$' and '$A\,\&\,(B \vee C)$', which are *distinct* LSL sentences.
- The term "*well-formed formula of LSL*" ("LSL WFF") is synonymous with "*LSL sentence*." Non-well-formed strings of symbols aren't sentences.
- In English, the string of English words 'Porch on the is cat a there' is ungrammatical — it is *not well-formed*. All of its *constituent parts* are English words/letters, but (*as a whole*) it's not an English sentence.
- Similarly, in LSL, the following strings of symbols are not WFFs:

'$A \to \vee B$' '$A\,\&\,B \vee C$' '$A \to B \to C$' '$\sim \vee B(\vee C)$' '$A\,\&\,B\,\&\,C$'

English ↦ LSL VI: Negation, Conjunction, and Disjunction

- The tilde '~' operates *only* on the unit that *immediately* follows it. In '$\sim K \vee M$,' ~ affects only 'K'; in '$\sim(K \vee M)$,' ~ affects the entire '$K \vee M$'.
- 'It is not the case that K or M' is *ambiguous* between '$\sim K \vee M$,' and '$\sim(K \vee M)$.' **Convention**: 'It is not the case that K or M' ↦ '$\sim K \vee M$'.
- 'Not both S and T' ↦ '$\sim(S\,\&\,T)$'. [Chapter 3: '$\sim(S\,\&\,T)$' *means the same as* '$\sim S \vee \sim T$'. But, '$\sim(S\,\&\,T)$' does *not* mean the same as '$\sim S\,\&\,\sim T$'.]
- 'Not either S or T' ↦ '$\sim(S \vee T)$'. [Chapter 3: '$\sim(S \vee T)$' *means the same as* '$\sim S\,\&\,\sim T$', but '$\sim(S \vee T)$' does *not* mean the same as '$\sim S \vee \sim T$'.]
- Here are some examples involving ~, &, and ∨ (not, and, or):
  1. Shell is not a polluter, but Exxon is. ↦ ??
  2. Not both Shell and Exxon are polluters. ↦ ??
  3. Both Shell and Exxon are not polluters. ↦ ??

4. Not either Shell or Exxon is a polluter. ↦ ??
5. Neither Shell nor Exxon is a polluter. ↦ ??
6. Either Shell or Exxon is not a polluter. ↦ ??

- Summary of translations involving ~, &, and ∨ (not, and, or):

"Logish"	**LSL**
Not either A or B.	$\sim(A \vee B)$
Either not A or not B	$\sim A \vee \sim B$
Not both A and B.	$\sim(A\,\&\,B)$
Both not A and not B. (Neither A nor B.)	$\sim A\,\&\,\sim B$

- DeMorgan Laws (we will *prove* these laws is Chapters 3 & 4):

  ⌜$\sim(p \vee q)$⌝ is equivalent to (means the same as) ⌜$\sim p\,\&\,\sim q$⌝
  ⌜$\sim(p\,\&\,q)$⌝ is equivalent to (means the same as) ⌜$\sim p \vee \sim q$⌝

- But, ⌜$\sim(p \vee q)$⌝ is *not* equivalent to ⌜$\sim p \vee \sim q$⌝.
- And, ⌜$\sim(p\,\&\,q)$⌝ is *not* equivalent to ⌜$\sim p\,\&\,\sim q$⌝.

English ↦ LSL VII: Summary of the LSL Connectives

English Expression	LSL Connective
not, it is not the case that, it is false that	~
and, yet, but, however, moreover, nevertheless, still, also, although, both, additionally, furthermore	&
or, unless, either … or …	∨
if … then …, only if, given that, in case, provided that, on condition that, sufficient condition, necessary condition, unless (**Note**: don't confuse antecedents/consequents!)	→
if and only if (iff), is equivalent to, sufficient and necessary condition for, necessary and sufficient condition for	↔

English ↦ LSL X (&, →): Example #1

- 'John will study hard and also bribe the instructor, and if he does both then he'll get an "A", provided the instructor likes him.'
  - Step 0: Decide on atomic sentences and letters.
    S: John will study hard. A: John will get an "A".
    B: John will bribe the instructor. L: The instructor likes John.
  - Step 1: Substitute into English, yielding "Logish":
    S and B, and if S and B then A, provided L.
  - Step 2: Make the transition into LSL (in stages as well, perhaps):
    S and B, and if L, then if S and B then A.
    $(S \,\&\, B) \,\&\, (L \rightarrow (\text{if } S \text{ and } B \text{ then } A))$.

    Final Product: $(S \,\&\, B) \,\&\, (L \rightarrow ((S \,\&\, B) \rightarrow A))$

English ↦ LSL II (~, &, ∨, →, ↔): Example #2

- 'Sara is going unless either Richard or Pam is going, and Sara is not going if, and only if, neither Pam nor Quincy are going.'
  - Step 0: Decide on atomic sentences and letters.
    P: Pam is going. Q: Quincy is going.
    R: Richard is going. S: Sara is going.
  - Step 1: Substitute into English, yielding "Logish":
    S unless either R or P, and not S iff neither P nor Q.
  - Step 2: Make the transition into LSL (in stages again):
    S unless $(R \vee P)$, and $\sim S$ iff $(\sim P \,\&\, \sim Q)$
    $(\sim(R \vee P) \rightarrow S) \,\&\, (\sim S \leftrightarrow (\sim P \,\&\, \sim Q))$
- It is also acceptable to replace the 'unless' with '$\vee$', yielding:
  $(S \vee (R \vee P)) \,\&\, (\sim S \leftrightarrow (\sim P \,\&\, \sim Q))$