Symbolic Logic Outline

What is Logic?

"Logic is the study of the methods and principles used to distinguish good (correct) from bad (incorrect) reasoning."

Irving M. Copi, *Introduction to Logic*, 6th ed., Macmillan Publishing Co., New York, 1982, p. 3.

How Do We Use Logic?

Every day, we put logic to work in making decisions about our lives, such as:

- How to dress?

  For example:

  Will it be hot or cold?

- What to eat and drink?

  For example:

  Will we need caffeine to stay up studying?

- Where to go?

  For example:

  Is it a Monday, in which case I need to go to CS1313?

Logical Inferences #1

We make logical ***inferences*** to reason about the decisions we need to make:

- It's cold this morning, so I need to wear a sweatshirt and jeans, not just a t-shirt and shorts.
- I've got a big exam tomorrow that I haven't studied for, so I'd better drink a couple pots of coffee.
- It's Monday, so I'd better be on time for CS1313, so that I'm on time for the quiz.

Logical Inferences #2

We can even construct more complicated chains of logic:

1. I have a programming project due soon.
2. I have been putting off working on it.
3. Therefore, I must start working on it today.

Symbolic Logic #1

In logic as in many topics, it sometimes can be easier to manage the various pieces of a task if we represent them **symbolically**.

- Let **D** be the statement
  "I have a programming project due soon."
- Let **L** be the statement
  "I have been putting off working on my programming project."
- Let **W** be the statement
  "I must start working on my programming project today."

We can then represent the chain of logic like so:

D and **L** => **W**

Symbolic Logic #2

D and **L** => **W**

This can be read in two ways:

- "**D** and **L** ***implies*** **W**."
- "**If** **D** is true **and** **L** is true, **then** **W** is true."

What If a Premise is False? #1

D and **L** => **W**

What if **L** is <u>**not**</u> true?

What if I've already started working on my programming project?

In that case, the statement

"I have been putting off working on my programming project"

is not true; it is <u>**false**</u>.

So then the statement

D and **L**

is also false. <u>**Why?**</u>

What If a Premise is False? #2

D and **L** => **W**

If the statement **L** is **false**, then why is the statement "**D** and **L**" also **false**?

Well, in this example, **L** is the statement "I have been putting off working on my programming project."

If this statement is false, then the following statement is true: "I **haven't** been putting off working on my programming project."

In that case, the statement **W** – "I must start working on my programming project today." – **cannot** be true, because I've **already** started working on it, so I can't **start** working on it now.

What If a Premise is False? #3

D and **L** => **W**

What if **D** is **false**?

What if I **don't** have a programming project due soon?

Well, statement **D** is:

"I have a programming project due soon."

So if I **don't** have a programming project due soon, then statement **D** is **false**.

In that case, statement **W** –

"I must start working on my programming project today."

– is also **false**, because I **don't** have a programming project due soon, so I **don't** need to start working on it today.

What If Both Premises are False?

D and **L** => **W**

What if **both** **D** and **L** are **false**?

In that case, I **don't** have a programming project due soon, and I've **already** gotten started on the one that's due in, say, a month.

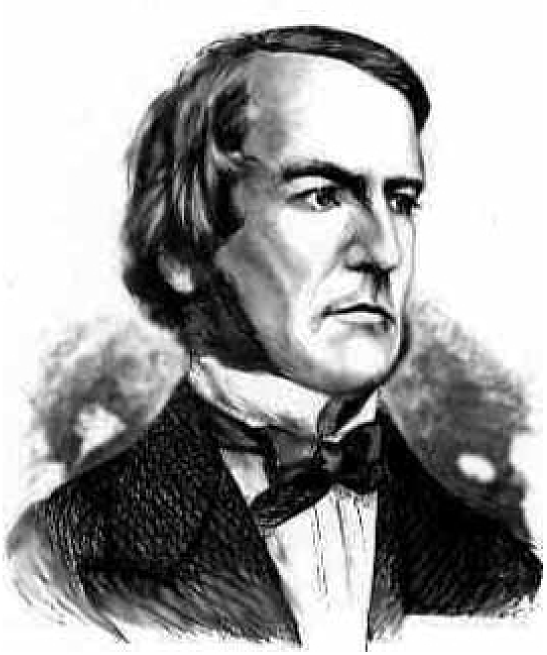
So I definitely **don't** need to start working on it today.

Boolean Values #1

A ***Boolean*** value is a **value** that is **either true or false**.

The name **Boolean** comes from George Boole, one of the 19th century mathematicians most responsible for formalizing the rules of symbolic logic.

So, in our example, statements **D**, **L** and **W** all are Boolean statements, because each of them is either true or false – that is, the **value** of each statement is either true or false.



http://thefilter.blogs.com/photos/uncategorized/boole.jpg

Boolean Values #2

D and **L** => **W**

We can express this idea symbolically; for example:

D = true

L = false

W = false

Note that

L = false

is read as "The statement **L** is false."

Boolean Values #3

L = false

is read as "The statement **L** is false."

In our programming project example, this means that the statement
"I **have** been putting off working on my programming project"
is **false**, which means that the statement
"**It is not the case that I have** been putting off working on my programming project"
is **true**, which in turn means that the statement
"I **haven't** been putting off working on my programming project"
is **true**.

So, in this case, "**L** = false" means that I **already have started** working on my programming project.

The AND Operation

From this example, we can draw some general conclusions about the statement

"**S1** and **S2**"

for **<u>any</u>** statement **S1** and **<u>any</u>** statement **S2**:

- If **S1** is **<u>true</u>** and **S2** is **<u>true</u>**, then "**S1** and **S2**" is **<u>true</u>**.
- If **S1** is **<u>false</u>** and **S2** is **<u>true</u>**, then "**S1** and **S2**" is **<u>false</u>**.
- If **S1** is **<u>true</u>** and **S2** is **<u>false</u>**, then "**S1** and **S2**" is **<u>false</u>**.
- If **S1** is **<u>false</u>** and **S2** is **<u>false</u>**, then "**S1** and **S2**" is **<u>false</u>**.

Truth Table for AND Operation

"**S1** and **S2**"

We can represent this statement with a ***truth table***:

		S2	
	AND	**true**	**false**
S1	**true**	true	false
	false	false	false

To read this, put your left index finger on the value of statement **S1** (that is, either true or false) at the left side of a row, and put your right index finger on the value of statement **S2** at the top of a column. Slide your left index finger rightward, and slide your right index finger downward, until they meet. The value under the two fingers is the value of the statement "**S1** and **S2**."

Another Boolean Operation

Suppose you want to know whether today is a good day to wear a jacket. You might want to come up with rules to help you make this decision:

- If it’s raining in the morning, then I’ll wear a jacket today.
- If it’s cold in the morning, then I’ll wear a jacket today.

So, for example, if you wake up one morning and it’s cold, then you wear a jacket that day.

Likewise, if you wake up one morning and it’s raining, then you wear a jacket that day.

Joining the Premises Together

We can construct a **general** rule by joining these two rules together:

If it's raining in the morning

OR

it's cold in the morning,

THEN

I'll wear a jacket today.

More on OR

We can apply symbolic logic to this set of statements, like so:

- Let **R** be the statement
  "It's raining in the morning."
- Let **C** be the statement
  "It's cold in the morning."
- Let **J** be the statement
  "I'll wear a jacket today."

We can then represent the chain of logic like so:

R or C => J

This can be read in two ways:

- "**R** or **C** implies **J**."
- "If **R** is true or **C** is true, then **J** is true."

What If a Premise is False? #1

What if **C** is **not** true?

For example, what if it's **hot** in the morning?

In that case, the statement
"It's cold in the morning"
is not true; it is false.

So then what about the statement "**R** or **C**"?

Well, even if it's hot in the morning,
if it's raining then you want your jacket anyway.

In other words, **if R is true**, then even though **C is false**,
still **"R or C" is true**.

What If a Premise is False? #2

Suppose that it's not raining in the morning, but it is cold.

Then the statement

"It's raining in the morning."

is **false**,

and the statement

"It's cold in the morning."

is **true**.

In which case, the statement

"I'll wear a jacket today."

is **true**.

In other words, if **R** is **false** and **C** is **true**, then

"**R** or **C**" is also **true**.

What If Both Premises are False?

What if **both** **R** and **C** are **false**?

In that case, it's neither raining nor cold in the morning, so I won't wear my jacket.

In other words, if **R** is **false** and **C** is **false**, then "**R** or **C**" is **false**.

The OR Operation

From this example, we can draw some general conclusions about the statement

"**S1** or **S2**"

for **<u>any</u>** statement **S1** and **<u>any</u>** statement **S2**:

- If **S1** is **<u>true</u>** and **S2** is **<u>true</u>**, then "**S1** or **S2**" is **<u>true</u>**.
- If **S1** is **<u>false</u>** and **S2** is **<u>true</u>**, then "**S1** or **S2**" is **<u>true</u>**.
- If **S1** is **<u>true</u>** and **S2** is **<u>false</u>**, then "**S1** or **S2**" is **<u>true</u>**.
- If **S1** is **<u>false</u>** and **S2** is **<u>false</u>**, then "**S1** or **S2**" is **<u>false</u>**.

Truth Table for OR Operation

"**S1** or **S2**"

We can represent this statement with a ***truth table***:

		S2	
	OR	**true**	**false**
S1	**true**	true	true
	false	true	false

To read this, put your left index finger on the value of statement **S1** (that is, either true or false) at the left side of a row, and put your right index finger on the value of statement **S2** at the top of a column. Slide your left index finger rightward, and slide your right index finger downward, until they meet. The value under the two fingers is the value of the statement "**S1** or **S2**."

Boolean OR is Inclusive

In symbolic logic, the Boolean operation OR is ***inclusive***, meaning that it can be the case that both statements are true.

In the jacket example, if it's raining **and** it's cold, then you'll take your jacket.

So Boolean OR is equivalent to **"and/or"** in normal colloquial speaking.

What is Exclusive OR?

We know that the Boolean **OR** operation is **inclusive**.

But, there's also such a thing as ***exclusive OR***, denoted **XOR**.

XOR is like OR, except that if **both** statements are **true**, then the **result is false**.

We **WON'T** be worrying about XOR in this course.

The NOT Operation

Boolean logic has another very important operation: ***NOT***, which changes a true value to false and a false value to true.

In real life, you've probably said something like this:

"I care what you think – **NOT**!"

Notice that the **NOT** exactly negates the meaning of the sentence: the sentence means "I don't care what you think."

From this example, we can draw some conclusions about the statement "not **S**," for **any** statement **S**:

- If **S** is true, then "not **S**" is false.
- If **S** is false, then "not **S**" is true.

Truth Table for NOT Operation

"NOT S"

We can represent this statement with a ***truth table***:

	S	
NOT	**true**	**false**
	false	true