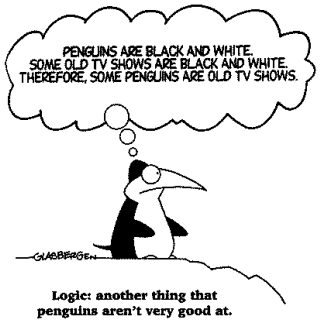
CS304 / CS310

Mathematical and Logical Aspects of Computing

Lecture 1: Tuesday, 4th September 2012



Logic: another thing that penguins aren't very good at.

Outline

Welcome to "Logic"

This course is called

Mathematical And Logical Aspects Of Computing

But usually, we'll just call it "*Logic*".
It is (at least) two module codes:

CS304 when taken by Science students

CS310 when taken by Arts students

Lecturer: Dr Niall Madden, School of Maths.
Room ADB-1013, Arás de Brún
Email: Niall.Madden@NUIGalway.ie, Phone (091 49) 3803.

Welcome to "Logic"

Lectures: Tuesday at **12.10** in the **AM150** and Friday at **10.10** in C219 (= ADB 1020). *?*

Tutorials: Combination of office hours and problem sessions, as well as a revision class before the final exam.

Web site: The on-line resources for this course are at `http://NUIGalway.BlackBoard.com` and `http://www.maths.nuigalway.ie/CS304`. There you'll find various pieces of information, including these notes. If you are not automatically enrolled onto CS304, please
- Try to enroll yourself. The required access code is:
- Failing that, send me an email, including your ID number.

You need to enroll on blackboard to get announcements, emails, and assessment results.

Assessment: Some homework exercises, and written **2 hour** exam at the end of Semester **1**.

Welcome to "Logic" — Topics

The central themes of CS304 include

1. **Propositional logic**:
   - How to give a precise mathematical formulation of logical statements;
   - How to determine if two statements are equivalent;
   - How to establish of two statements are consistent, i.e., don't contradict each other.
   - Validity of arguments.
   - Boolean algebra.
2. **Predicate calculus**:
   - The limitations of propositional logic;
   - Quantifers (existential and universal);
   - Semantic entailment; Resolution....

Welcome to "Logic" — Text book

There is no required textbook for CS304, but I will recommend a few for particular topics, including

- Mordechai Ben-Ari, *Mathematical Logic for computer science*. (511.3 BEN)
- John Kelly, *Essence of Logic*. (511.3 KEL)
  Also:
- Stefan Waner and Steven R. Costenoble, *Introduction to Logic*, http://tinyurl.com/IntroToLogic
- Ian Chiswell and Wilfred Hodges, *Mathematical Logic* (511.3 CHI).
- Kenneth Rosen, *Discrete Mathematics and its applications* (511.1 ROS).
- Huth and Ryan, *Logic in Computer Science*, 005.1015113 HUT.
- S.N. Burris *Logic of Mathematics and Computer Science*

Boolean Operators

We are familiar with many different sets, and the basic operators that can be applied to them. [*EXAMPLES*].

For this course we are particularly interested in:

- The set denoted $\{F, T\}$ ("false" and "true")
  and
- the basic operators that we can apply to the two elements of the set above: usually just one or two (later more).

George Boole (1815–1864)
first Professor of Mathematics at UCC.

Boolean Operators — 1-place operators

There are **4** possible operators that take a single argument. They are:

If these, the 1st and 4th are trivial, the 2nd is the *identity operator* and the 3rd is called *negation*, denoted by $\neg$, and read as "not".

Boolean Operators — 2-place operators

There are **16** possible operators that take a two argument. Here are examples of 6 of these:

Of these the most important, other than the identity are

- *Conjunction*, written as $a \wedge b$, read as "and".
- *Disjunction*, written as $a \vee b$, read as "or".

But others are important too – we'll come back to them later.

Boolean Operators — The conjunction operator

The conjunction operator ("and"), $\wedge$, is probably the simplest 2-place operator, given that it agrees with it usage of the word "and" in natural language. Here is the table again:

[EXAMPLE:] Let's consider the "proposition"

Today is Tuesday **and** *today we have a Logic lecture*

...(take notes)

Boolean Operators — The disjunction operator

The *disjunction operator* ("or"), $\vee$, is slightly more subtle, since it does not exactly agree with how we often use the word in natural language.

[EXAMPLE:] Let's consider the "proposition"

Today is Wednesday **or** *today we have a Logic lecture*

...(take notes ... it helps to consider under what conditions would this statement be true, and under what conditions would it be false.)

Boolean Operators — Non-equivalence

Finally, consider the proposition:

I am Arts student or a Science student

If we were to interpret this in natural language the "truth table" would be:

This is called the "*non-equivalence*", or, more commonly, the *exclusive or* operator, and denoted as $\oplus$