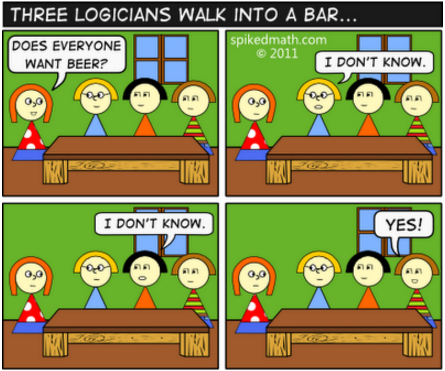
CS304/CS310
Mathematical and **Logic**al
Aspects of Computing

Lecture 4: Venn diagrams
and Disjunctive Normal Form

Friday,
13^{th} Sep 2013



Today's topics:

1. The implication and Wason's selection task (again)
2. Venn diagrams
3. (Disjunctive) normal form

. . . .

Announcement: tutorials start next week. Wednesday at 5pm in AM112.

Towards the end of Lecture 3, we introduced the *implication* operator, which is written as $a \rightarrow b$, and read as "a implies b". (See also Example 2.21 of Ben-Ari). The table is:

We also saw that there are several other ways of expressing this operator:

(a) $a \rightarrow b \equiv (\neg a) \vee b$ (We did this one in class);

(b) $a \rightarrow b \equiv \neg(a \wedge \neg b)$

Now we'll check (b) by writing out the truth table for $(a \rightarrow b) \leftrightarrow \neg(a \wedge \neg b)$.

(See Chapter 4 of Chiswell and Hodges "*Mathematical Logic*")
You will be shown 4 cards. Each has a number written on one side and a letter on the other.

You are shown a statement S about the cards. Then you must answer the following question:

> *"Which card or cards must I turn over in order to check whether the statement S is true?"*

The cards are:

The statement is:

> S: *If a card has a vowel one on side, it has an even number on the other.*

There is a close relationship between the study of logic and set theory.
As mentioned last week, the *conjunction ($\wedge$)* operator in logic is related to *intersection ($\cap$)* in set theory. This can be demonstrated using a Venn diagram.

Example:

Similarly, there is a relationship between

- *disjunction ($\vee$)* and *union ($\cup$)*.
- *negation* and *compliment*

Example:

We can also use **Venn diagrams** as a way of establishing logical equivalences. Consider the diagrams for

(a) $a \leftrightarrow b$

(b) $(a \longrightarrow b) \wedge (b \longrightarrow a)$.

So far we have been able to describe every Boolean Expression in terms of a logic table (equivalently: given an interpretation for all possible assignments). This begs the question:

Does every logic table have a Boolean expression?

This can be answered (in the affirmative) is using something called *disjunctive normal form* (DNF). First some definitions:

- A *normal form* of a Boolean expression uses only the operators: $\{\wedge, \vee, \neg\}$.
- A *literal* is an atom or its negation.
- **Disjunctive Normal Form** (DNF) is the *disjunction of conjunction of literals*, and is the topic to today's lecture.
- **Conjunctive Normal Form** (CNF) is the *conjunction or disjunction of literals*, and will feature (a lot) later in this course.

Examples:

Method for constructing the Disjunctive Normal Form of a logic table

1. For each row (i.e., assignment) with output (i.e., interpretation) of **T**, write down the formula that uses only **AND** and **NOT** and has interpretation **T** only for that assignment.
2. Write down the disjunction of these formulae (i.e., using "OR").

Example: Write the Boolean formula, defined by the following table, in DNF.

a	b	c	E
F	F	F	T
F	F	T	F
F	T	F	F
F	T	T	T
T	F	F	F
T	F	T	T
T	T	F	F
T	T	T	F

1. Convince yourself that the Boolean expressions $\mathcal{E}_1$ and $\mathcal{E}_2$ are logically equivalent if, and only if, the expression $\mathcal{E}_1 \leftrightarrow \mathcal{E}_2$ evaluates as **true** for all possible interpretations. (We call this a *tautology*: more of this later).
2. For each of the following expressions, state if they are in DNF, CNF or neither:
   (a) $(a \wedge \neg b) \vee (c \wedge \neg a)$
   (b) $a \wedge (b \vee c)$
   (c) $a \wedge \neg(b \vee c)$
   (d) $a \wedge (\neg b \wedge \neg c)$
   (e) $\neg a \vee b$