CS/Math 240: Introduction to Discrete Mathematics	1/20/2010
Lecture 2 : Propositions	
Instructor: Dieter van Melkebeek	Scribe: Dalibor Zelený

DRAFT

Last time we analyzed various maze solving algorithms in order to illustrate some of the techniques covered in this course. We did not expect you to understand everything from that lecture. We did not prove everything as formally as we should have, either. From now on, we will be more rigorous, and you should ask a question whenever you don't understand something.

2.1 Propositions and Proofs

Remember that the goal of this course is to teach you how to reason about discrete structures in a rigorous manner. To that end, we discuss propositions in this lecture. Propositions are clear statements that are either true or false.

Definition 2.1. *A* proposition *is a statement that is either true or false.*

Definition 2.2. *A* proof *of a proposition is a chain of logical deductions ending in the proposition and starting from some axioms.*

We make a few remarks about Definition 2.2. First, think of logical deductions as steps that are logically sound. Second, axioms are statements we take for granted and, therefore, don't need to prove. The set of axioms depends on the area we work in. For example, when we talk about Euclidean geometry, there are five basic axioms. The first axiom of Euclidean geometry states that we can draw a straight line segment connecting any two points.

2.1.1 Proofs "By Picture"

A common proof technique is to capture a proposition using descriptive pictures and then reason about the pictures. This is a very powerful technique as it allows us to use our intuition. It is said that "a picture is worth a thousand words"; however, be warned that in some cases our intuition may lead us astray.

We now give some proofs by picture as examples. Our first example is a proof of the Pythagorean Theorem. Our second example is a "proof" by picture that leads to an incorrect conclusion.

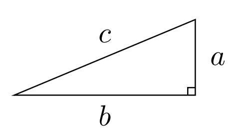
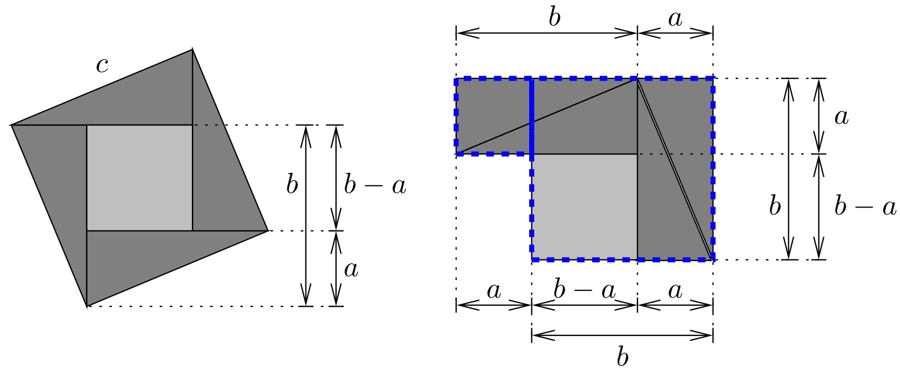


Figure 2.1: A right triangle with sides of length a, b, and c

Proposition 2.3 (Pythagorean Theorem)**.** *In a right triangle where the hypotenuse has length c and the other two sides have lengths a and b, we have $a^2 + b^2 = c^2$.*

Proof. Consider a right triangle like the one in Figure 2.1. We take four copies of the triangle and arrange them in two different ways.

First, form a square with the hypotenuse as the side. Its area is c^2. We show this arrangement in Figure 2.2a. The four copies of the right triangle are shaded in dark gray. The space shaded light gray inside the square in Figure 2.2a is a square of side length $b - a$.



(a) An arrangement of four copies of the right triangle from Figure 2.1. There is a square in the middle.

(b) A rearrangement of Figure 2.2a.

Figure 2.2: Two arrangements of four triangles and a square.

Now rearrange the five pieces differently, as shown in Figure 2.2b. The thick blue lines indicate that we can view this arrangement as two squares of sides a and b placed next to each other. The square on the right has length is b, so the area of that square is b^2. The square on the left has side length a, so its area is a^2, and the total area is, therefore, $a^2 + b^2$.

Since we obtained the second picture from the first one by rearranging, they have the same area, which completes the proof that $a^2 + b^2 = c^2$. □

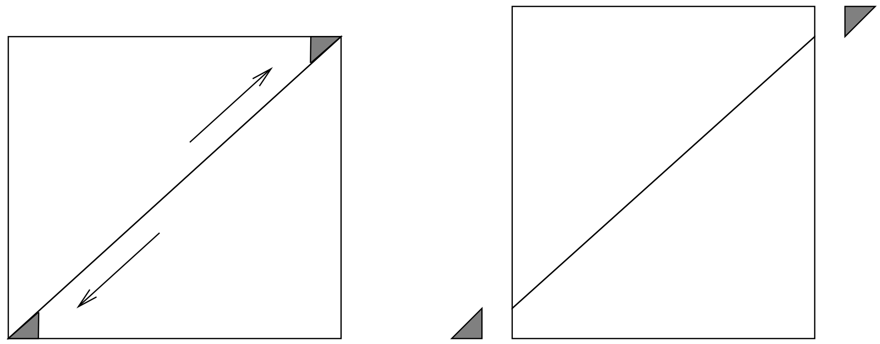
Now let's see a "proof" of a false statement. It looks very similar to the proof of the Pythagorean Theorem we just did, but has a mistake in it. Take the next example as a warning that you have to be careful when you write proofs.

Incorrect proof that $1 = 2$.

Two right triangles with sides 10 and 11 form a rectangle with area $10 \cdot 11 = 110$. The situation is shown in Figure 2.3a. Take out small pieces from the corners. As we can see from the picture, they are right triangles whose two shorter sides both have length 1. Now slide the two remaining parts of the triangles together so that they form a rectangle again. The new rectangle has side lengths 10 and 11, and, thus, an area of $10 \cdot 11 = 110$. We also have two extra pieces with a total area of 1. This is shown in Figure 2.3b.

Since we obtained Figure 2.3b from Figure 2.3a, both pictures have the same area, which implies that $110 = 111$. Subtracting 109 from both sides gives us $1 = 2$. □

The issue is that we drew a misleading picture in Figure 2.3a. The picture looks like a square, which made us think that the two shorter sides of our small triangles we were cutting out both had length 1. But that is not true. When one side has length 1, the other side has length either 11/10 or 10/11.



(a) Arrows show how to slide the remaining parts of our two triangles after cutting out the shaded corner pieces.

(b) A rearrangement of the picture from Figure 2.3a.

Figure 2.3: A picture for an incorrect proof that $1 = 2$.

2.1.2 More on Propositions

Recall from Definition 2.1 that a proposition is a statement that is either true or false. Let us see some examples of propositions.

Example 2.1:

- P_1: "Madison is the capital of Wisconsin." - this is a true proposition
- P_2: "The Yahara river flows into Lake Michigan." - this is a false proposition

⊠

Now let's see some statements that are not propositions

Example 2.2:

- "What is the capital of Wisconsin?" - This is a question, not a statement.
- "Is Madison the capital of Wisconsin?" - That's a little better. This is a yes/no question; nonetheless, it is still a question and not a statement.
- "This sentence is not true." - This is not a valid proposition. We cannot assign a truth value to this statement.

⊠

The last sentence is a statement to which we cannot assign a truth value. Assume the sentence were true. Then the statement would say that the sentence is false, which is a contradiction because we said it was true. On the other hand, suppose the statement were false. Then the sentence would be true, and we have a contradiction again. The last sentence is a self-referential statement that is contradictory. It is also connected to the halting problem, and we may say something about that later in the course.

We will not deal with statements like "this sentence is false" in this course. Let's look at some statements that we are actually going to see in this course.

Example 2.3:

- G: Every even integer larger than 2 can be expressed as the sum of two primes.

⊠

Recall that primes are positive integers greater than one that are only divisible by 1 and themselves. The first few primes are $2, 3, 5, 7, 11, \ldots$. In order to argue that the statement G is false, we would need to find an even integer that cannot be written as a sum of two primes. To show that this statement is true, we would need to argue about infinitely many integers. It is actually not known whether G is a true statement or not. It is called *Goldbach's conjecture*. People have tried to find a proof as well as a counterexample, but have not succeeded yet.

To start verifying Goldbach's conjecture, we could start writing even numbers and finding two primes for each of them. For example, $4 = 2+2$, $6 = 3+3$, and $8 = 3+5$. (Note that the statement doesn't say anything about repeating primes, which is why we can write $4 = 2 + 2$.) But there are infinitely many even integers greater than 2, so we would never complete such a proof.

Now consider the following modification of G: "Every integer greater than 3 can be expressed as a sum of two primes". This is a false statement. It may not seem that way at first because it holds for all integers up to 10. But we cannot write 11 as a sum of two primes.

Notice that 11 is an odd number. If we want to break any odd number n into two integers that add up to n, one of the two integer pieces must be even and the other one must be odd because that's the only way how we can get an odd integer as a sum of two integers. Since we want both of the integer pieces to be prime, the even piece must be 2 because 2 is the only even prime. But if we want the two pieces to add up to 11, this forces the other piece to be 9, which is not prime. Hence, we cannot write 11 as a sum of two primes, and our modification of G is false.

2.2 Operations on Propositions

By an operation on propositions, we mean that we take one or more propositions and combine them to get a new proposition. In English, we use words such as "not", "and", "or", "if . . . then" to do so. The mathematical symbols for these are $\neg$, $\wedge$, $\vee$ and $\Rightarrow$, respectively. In a programming language, we would use the symbols !, &&, ||, and if statements, respectively.

We need to be careful when going from English to the language of mathematics or to a programming language, however. The word "or" don't translate exactly into $\vee$. For example, when we say "you can have cake or you can have ice cream", we usually give the person we talk to an option to take one or the other, but not both. That is, the meaning is exclusive. But to a mathematician, "or" is not exclusive, so a mathematician could take both cake and ice cream.

There is a similar issue with the if-then construct. Suppose a father tells his son: "If you graduate with a 4.0 GPA, I will buy you a car". The intended meaning is that the son will get the car only if he gets a 4.0. As we will see, a child who is a mathematician can hope to get a car even without a 4.0 GPA.

2.2.1 Overview of Operations

First let's view the word "not" as the mathematical operation $\neg$. If P is a proposition, we read $\neg P$ as "*not P*". No matter what P is, $\neg P$ is true when P is false, and $\neg P$ is false when P is true. We capture this notion in the form of a *truth table*. See the truth table of the $\neg$ operator in Table 2.1. In all tables below, T stands for "true" and F stands for "false".

Now let's see the meanings of "and", "or", and "if...then" in the language of mathematics. If P and Q are propositions, we read $P \wedge Q$ as "*P and Q*", $P \vee Q$ as "*P or Q*", and $P \Rightarrow Q$ as "*P*

P	$\neg P$
T	F
F	T

Table 2.1: The truth table of "not" ($\neg$)

implies Q". In the *implication* $P \Rightarrow Q$, we call P the *premise* and Q the *consequence*. We list the truth tables of the three operators we just discussed together in Table 2.2.

P	Q	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$
T	T	T	T	T
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

Table 2.2: Truth tables of "and" ($\wedge$), "or" ($\vee$), and "implies" ($\Rightarrow$)
.

We see from the truth table of $\vee$ why a mathematician can get both cake and ice cream. The row in the truth table of $P \vee Q$ where P and Q are both true has a truth value of T. Hence, if P stands for "you get cake" and Q stands for "you get ice cream", taking both makes the sentence "you get cake or you get ice cream" true, which means taking both is a valid option.

Now let's see why children who study mathematics can hope to get a car even if they don't get a 4.0 GPA. We see from the truth table that the only way the statement $P \Rightarrow Q$ can be false is if the premise "you have a 4.0 GPA" is true and the consequence "you get a car" is false. Thus, if the premise is false, the consequence can be either true or false for the implication to be true.

2.2.2 More Examples of Implications

Let's recall some propositions we stated earlier in this lecture.

- P_1: Madison is the capital of Wisconsin.
- P_2: The Yahara river flows into Lake Michigan.
- G: Every even integer larger than 2 can be expressed as the sum of two primes.

Example 2.4: Consider the implication $G \Rightarrow P_1$. In English, this says "If Goldbach's conjecture is true, then Madison is the capital of Wisconsin." This is a true proposition. We would be tempted to say that the proposition is false because we don't know whether G holds or not. But we know that Madison is the capital of Wisconsin, and both rows of the truth table of an implication where the consequence is true have T as the truth value (see Table 2.2 with $P = G$ and $Q = P_1$). Thus, regardless of the truth of Goldbach's conjecture, the implication $G \Rightarrow P_1$ is true. ⊠

Example 2.5: Consider the following proposition: "If P_2 then $1 = 2$". This is a true proposition. The premise P_2 is false and the consequence is false, so the implication is true. ⊠

Example 2.6: "If P_1 then $1 = 2$" is a false proposition. The premise is true, the consequence is false, and the truth table of the implication for this case says "false". Let us stress once again that this is the only way in which an implication can be false. ⊠

In the spirit of the last example, we may be tempted to "prove" Goldbach's conjecture by saying that "If $1 = 2$ then Goldbach's conjecture holds." However, this does not constitute a valid proof. This only gives us a true implication. For a proof to be valid, it does not suffice to give a true implication. We also need to show that the premise of that implication is true in order to get a valid proof. We will discuss this in more detail later.

2.2.3 Combining Propositions

Once we have operators, we can start combining them to obtain propositional formulas such as $(P \vee (\neg P \wedge Q))$.

Definition 2.4. *A* propositional formula *is a proposition obtained by applying a finite number of operators ($\neg$, $\wedge$, $\vee$, $\Rightarrow$) to propositional variables ($P, Q, \ldots$).*

2.2.4 Logical Equivalence

It may be the case that multiple propositional formulas are the same.

Definition 2.5. *Let F_1 and F_2 be two propositional formulas. We say F_1 and F_2 are* logically equivalent *if they have the same truth value for all possible settings of the variables.*

Example 2.7: $P \vee (\neg P \wedge Q)$ is logically equivalent to $P \vee Q$.

There are many ways to see this. Since we are discussing truth tables, let us look at the truth tables of the two propositional formulas. If we compare the truth tables of the two propositional formulas row by row and find no difference, then, by Definition 2.5, the two formulas are logically equivalent. We show the truth tables together in Figure 2.3. Notice that the columns corresponding to $P \vee (\neg P \wedge Q)$ and $P \vee Q$ are the same, so the two formulas are indeed logically equivalent.

P	Q	$\neg P$	$\neg P \wedge Q$	$P \vee (\neg P \wedge Q)$	$P \vee Q$
T	T	F	F	T	T
T	F	F	T	T	T
F	T	T	T	T	T
F	F	T	F	F	F

Table 2.3: Showing that $P \vee (\neg P \wedge Q)$ and $P \vee Q$ are logically equivalent.

$\boxtimes$

Example 2.8: $P \Rightarrow Q$ is logically equivalent to $\neg Q \Rightarrow \neg P$.

For example, say $P =$ "it is snowing" and $Q =$ "it is cold". Then $P \Rightarrow Q$ says "If it is snowing, then it is cold", and $\neg Q \Rightarrow \neg P$ says "If it is not cold, it is not snowing". Intuitively, those two sentences say the same thing. Let's see the truth tables in Table 2.4 for a verification.

P	Q	$\neg P$	$\neg Q$	$\neg Q \Rightarrow \neg P$	$P \Rightarrow Q$
T	T	F	F	T	T
T	F	F	T	F	F
F	T	T	F	T	T
F	F	T	T	T	T

Table 2.4: Showing that $P \Rightarrow Q$ and $\neg Q \Rightarrow \neg P$ are logically equivalent.

⊠

We call $\neg Q \Rightarrow \neg P$ the *contrapositive* proposition of $P \Rightarrow Q$. This is an important concept for proofs. Since an implication is logically equivalent to its contrapositive, it suffices to prove the contrapositive in order to prove the implication. This is a common technique in writing proofs because the contrapositive may have a simpler or more intuitive proof than the implication.

An implication also has a converse. The *converse* of the implication $P \Rightarrow Q$ is the implication $Q \Rightarrow P$.

Example 2.9: The implication and its converse are not logically equivalent.

Look at Table 2.5 which shows the truth tables of the two implications. The two middle rows of those truth tables are different, so $P \Rightarrow Q$ and $Q \Rightarrow P$ are not logically equivalent.

P	Q	$P \Rightarrow Q$	$Q \Rightarrow P$
T	T	T	T
T	F	F	T
F	T	T	F
F	F	T	T

Table 2.5: Showing that an implication and its converse are not logically equivalent.

Consider P: "it is snowing" and Q: "it is cold". The implication $P \Rightarrow Q$ says that "if it is snowing, it is cold", while the implication $Q \Rightarrow P$ says that "if it is cold, it is snowing". The former doesn't allow the possibility of warm weather when it snows, while the latter does allow that. ⊠

2.2.5 The Equivalence Operator

There is another logical operator that we use in mathematics. We use it to denote that both an implication and its converse hold. We write this as $P \iff Q$ and read as "P *if and only if* Q". In other words, $P \iff Q$ is logically equivalent to $(P \Rightarrow Q) \wedge (Q \Rightarrow P)$.

Again recall the statements P_1 and P_2 from earlier. We use them in the example below.

Example 2.10:

- "$P_1 \iff P_2$" is a false statement because P_1 is true while P_2 is false.
- "$P_1 \iff 1 = 2$" is false for the same reason.
- "$P_2 \iff 1 = 2$" is true because both sides of the equivalence operator are false.
- "x is even $\iff$ $x + 1$ is odd" is true.

⊠

2.2.6 The P Versus NP Problem

Consider the following problem. Somebody gives you a propositional formula. Your goal is to set each variable in that formula to either true or false in a way that makes the entire formula true. This is known as the *satisfiability problem* and is equivalent to many problems in computer science and engineering.

One algorithm for satisfiability is exhaustive search. We try all possible settings to the variables, but this is an algorithm whose running time is exponential in the number of variables. To see that, consider a formula that has k variables. Each variable has two possible values, and each variable can be assigned a value independently, so we have to multiply together the numbers of possible assignments for each variable to get a total of 2^k possible assignments of truth values to the variables.

We would like to know whether there is a faster algorithm for deciding satisfiability, preferably one that runs in time that is polynomial in the number of variables. This is known as the P *versus* NP *problem*. We do not know whether a satisfiability algorithm that is substantially faster than exhaustive search exists. On the other hand, we are also unable to disprove the existence of such an algorithm, which is what makes the P versus NP problem one of the most important open problems in computer science and mathematics.

2.3 Next Time

Next time we will generalize propositions and talk about predicates.