CS/Math 240: Introduction to Discrete Mathematics

Reading 2 : Propositions and Predicates

Author: Dieter van Melkebeek (updates by Beck Hasti and Gautam Prakriya)

2.1 Propositions

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 some basic mathematical logic in this reading. We begin our discussion with propositions. Propositions are clear statements that are either true or false.

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

Let us look at 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 $2, 3, 5, 7, 11, \ldots$.

G is a declarative statement that is either true or false and is therefore a proposition. This proposition is called *Goldbach's conjecture*. It is not known whether Goldbach's conjecture is true or 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 parent tells their child: "If you graduate with a 4.0 GPA, I will buy you a car". The intended meaning is that the child will get the car only if a GPA 4.0 is obtained. 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".

P	$\neg P$
T	F
F	T

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

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 *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 2.2. *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 2.3. *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 2.3, the two formulas are logically

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.

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.

⊠

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 Truth of Propositions

We haven't yet addressed the question of how to determine if a proposition is true. The methodology in mathematics is as follows:

- Assume that certain propositions are true. (These assumptions are called axioms)
- Any statement that can be derived from the axioms via a sequence of logical deductions is understood to be true. Such a derivation is called a *proof.*

We make a few remarks about the above. 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, the first axiom of Euclidean geometry states that we can draw a straight line segment connecting any two points.

To reason about proofs, we will need to talk about groups of propositions. Predicates allow us to do this

2.3 Predicates

We begin with a definition.

Definition 2.4. *A* predicate *is associated with some underlying domain D, and we define it as a mapping from D to propositions.*

We can also view a predicate as a parametrized proposition, where the parameter ranges over the domain D. We now give some examples of predicates.

Example 2.11:

Over $D = \{0, 1, 2, \ldots\}$, consider the predicate Even(x): x is even.

- Even(2) means "2 is even", which is a true statement.
- Even(3) means "3 is even", which is a false statement.

⊠

Example 2.12:

Still over $D = \{0, 1, 2, \ldots\}$, consider the predicate $\text{Prime}(x)$: x is prime.

This predicate is true if x is prime, and false otherwise. That is, $\text{Prime}(3)$ is true, and $\text{Prime}(4)$ is false. ⊠

2.3.1 New Predicates From Old

We can compound predicates the same way we put together propositions. We can use the operators $\neg$, $\vee$, $\wedge$, $\Rightarrow$, and $\iff$ to form more complex predicates. There is also an additional operation defined on predicates called *quantification*. Quantification allows us to express concepts such as "all elements of D" or "some elements of D". The former is called *universal quantification* and the latter is called *existential quantification*.

2.3.2 Universal Quantification

For *universal quantification*, we use the notation $(\forall x)P(x)$ to mean "for all x, $P(x)$ holds". The statement $(\forall x)P(x)$ is true if $P(x)$ is a true proposition for every x in the domain. It is false if there is at least one x in the domain for which $P(x)$ is a false proposition.

Let's see an example and make a few remarks.

Example 2.13:

$(\forall x)\text{Even}(x)$ is a false proposition. To see this, we just need to find one x for which $\text{Even}(x)$ is false. One counterexample is $x = 1$.

$(\forall x)\text{Even}(x) \iff \neg\text{Even}(x + 1)$ is a true proposition. If x is even, then $x + 1$ is odd, and if x is odd, then $x + 1$ is even. Thus, the equivalence holds for every x. ⊠

What if our domain D is empty? Is the statement $(\forall x)P(x)$ true? We cannot find an element of the domain for which P doesn't hold because D is empty. Therefore, the statement is true.

In the examples above, we omitted the domain D. If we want to state the domain explicitly, we can say $(\forall x \in D)$ instead of just $(\forall x)$. It is common in practice to omit the domain when it is clear from the context what the domain is; however, in order to be precise, we should always specify the domain.

2.3.3 Existential Quantification

For *existential quantification*, we write $(\exists x)P(x)$ to mean "there exists an x for which P holds". The statement $(\exists x)P(x)$ is true if $P(x)$ is a true proposition for at least one x in the domain. It is false if there is no x in the domain for which $P(x)$ is a true proposition.

Again, we illustrate this on an example.

Example 2.14:

$(\exists x)\text{Even}(x)$ is a true proposition. To see this, we just need to find one x for which $\text{Even}(x)$ holds. For example $x = 0$ is an example of an x for which $\text{Even}(x)$ holds.

$(\exists x)\text{Even}(x) \wedge \text{Prime}(x) \wedge x > 2$ is false because all primes larger than 2 are odd. ⊠

Here we should also warn the reader that the use of the word "any" in English is ambiguous and we have to rely on the context in order to understand its meaning. Sometimes, this word is equivalent to "some", whereas in other situations it could be equivalent to "all". For example, consider the sentence: "If you can solve any homework problem, you will get an A." Almost certainly, "any" here means "all" and not "at least one", although most students would wish the opposite were true. When speaking in the language of mathematics, we should avoid using such words, and use unambiguous expressions such as "for all" or "there exists" instead.

2.3.4 Universal and Existential Quantifiers Are Related

There is a duality between existential and universal quantification. Consider the sentences: "Not everyone likes sunshine," and "Some people don't like sunshine." These two sentences say the same thing. To see that, let's write them in mathematical notation. The domain is the set of all people, and we have the predicate $S(x)$: x likes sunshine. The first sentence would then be

$$\neg(\forall x)S(x), \tag{2.1}$$

and the second sentence would be

$$(\exists x)\neg S(x). \tag{2.2}$$

In fact, no matter what the domain is and what S is, the statements (2.1) and (2.2) are logically equivalent. If it's not true that a proposition holds for every element of the domain, there must be an element of the domain for which the proposition doesn't hold. The converse of the last sentence is also true, i.e., if there is an element of the domain for which a proposition doesn't hold, it is not the case that the proposition holds for all elements of the domain. The statements $\neg(\exists x)S(x)$ and $(\forall x)\neg S(x)$ are logically equivalent for the same reason.

In the case of a statement with an existential quantifier, proving that it's false is harder than proving that it is true. In order to prove that such a statement is false, we need to show that the existentially quantified statement is false for every element of the domain, whereas to prove that it's true, we just need to find an element of the domain for which the existentially quantified statement holds. Observe that the exact opposite is true about universally quantified statements. Again, we see there is a duality.

There is a similar duality between conjunctions and disjunctions. We could write $\forall$ as a "big AND" and an $\exists$ as a "big OR". For example, we could rewrite the statement $(\forall x)\text{Prime}(x)$ as "2 is prime, and 3 is prime, and 4 is prime, and 5 is prime, ..." Putting a "not" in front of the sentence would give us a sentence of the form (2.1). We could then rewrite it in the form (2.2) as "Either 2 is not prime, or 3 is not prime, or 4 is not prime, or 5 is not prime, ..."

Another instance of the duality between conjunctions and disjunctions is *DeMorgan's law*, which states the following two logical equivalences.

- $\neg(P \wedge Q)$ is logically equivalent to $\neg P \vee \neg Q$.
- $\neg(P \vee Q)$ is logically equivalent to $\neg P \wedge \neg Q$.

2.3.5 Translating into Predicate Notation

Let D be the set of all humans. Consider the predicates $S(x)$: "x loves sunshine" and $W(x)$: "x is in Wisconsin".

Let's translate the English sentences: (i) "Everyone in Wisconsin loves sunshine." and (ii) "Someone in Wisconsin doesn't love sunshine." into predicate notation.

Sentence (i) translates to

$$(\forall x \in D)(W(x) \Rightarrow S(x)). \tag{2.3}$$

This statement says "For every human x, if x is in Wisconsin then x loves sunshine."

To check that this is the correct translation, we need to check that when statement (i) is true (2.3) is also true and when (i) is false, (2.3) is false.

Note that if x is not in Wisconsin, i.e. if $W(x)$ is false, then $W(x) \Rightarrow S(x)$ is true. On the other hand, if x is in Wisconsin, then $W(x)$ is true. In which case, the truth value of $(W(x) \Rightarrow S(x))$ is the same as the truth value of $S(x)$ (see the truth table for implication).

This tells us that if statement (i) is true, i.e. $S(x)$ is true for every person in Wisconsin, then (2.3) is true. And if (i) is false, i.e. there is some person in Wisconsin for whom $S(x)$ is false, then (2.3) is false.

Sentence (ii) translates to

$$(\exists x \in D)(W(x) \wedge \neg S(x)). \tag{2.4}$$

Again let's check that when statement (ii) is true (2.4) is true and when statement (ii) is false (2.4) is false

Note that $W(x) \wedge \neg S(x)$ is false if x is not in Wisconsin. And if x is in Wisconsin, i.e. $W(x)$ is true, then the truth value of $(W(x) \wedge \neg S(x))$ is the same as that of $\neg S(x)$.

Therefore, if there is a person in Wisconsin who doesn't love sunshine, then (2.4) is true. And if every person in Wisconsin loves sunshine, then (2.4) is false.

The use of implication with the universal quantifier ensures that the statement doesn't become false because of elements we don't care about, and the use of conjuntion with the existential quantifier ensures that the statement doesn't become true because of elements we don't care about.

2.3.6 Multiple Quantifiers and Domains

We will encounter statements that use both the existential and universal quantifiers together. Consider the statement,

$$(\forall x)(\exists y)(y = x + 1).$$

This statemet says "For every number x there is some number y such that $y = x + 1$".

When combining quantifiers the order in which the quantifiers appear is very important. Switching the order changes the meaning of the statement. For instance the statement

$$(\exists y)(\forall x)(y = x + 1).$$

says "There is some number y that is one more than every number x." This statement is clearly false.

Let us investigate the effect of switching quantifiers more closely. Consider the following statements:

1. $(\forall x)(\exists y)P(x, y)$. (For every x there is a y such that $P(x, y)$ holds.)
2. $(\exists y)(\forall x)P(x, y)$. (There is a y such that for every x $P(x, y)$ holds.)

Note that if 2 is true, 1 is true as well. Moving the universal quantifier to the left of an existential quantifier doesn't change a true statement into a false statement. But this comes at a price. We lose information when we perform such an operation because now we are saying "for every x there is some y.." as opposed to having a single y that works for all x.

On the other hand if statement 1 is true we can't conclude anything about statement 2, as we saw in the example above.

2.3.6.1 Translating Goldbach's Conjecture into predicates

Recall that Goldbach's conjecture states: "Every even integer greater than 2 can be written as a sum of two primes". There are actually two domains involved in the statement of Goldbach's conjecture, namely the even integers greater than 2 and the prime numbers. Moreover, there are multiple quantifiers present in the statement, which becomes apparent if we rephrase it as "*for all* even numbers x greater than two *there exist* two primes that add up to x".

Let us now rewrite the statement of Goldbach's conjecture using predicate notation. First, we need to give names to our domains. Let $D_1 = \{x \mid \text{Even}(x) \wedge x > 2\}$ be the set of even integers larger than 2, and $D_2 = \{x \mid x \text{ is an integer such that } x \text{ is prime}\}$ the set of primes. Now we can write Goldbach's conjecture as follows:

$$(\forall x \in D_1)(\exists y \in D_2)(\exists z \in D_2) x = y + z \tag{2.5}$$

Now let's write (2.5) using only one domain, $D = \{0, 1, 2, \ldots\}$, and our predicates Even and Prime. We first eliminate D_1. Start with the predicate

$$(\forall x \in D_1) P(x) \qquad \text{where} \quad P(x) : (\exists y \in D_2)(\exists z \in D_2) x = y + z. \tag{2.6}$$

Note that an integer x is in D_1 if x is even and greater than 2, i.e., if $\text{Even}(x) \wedge x > 2$, so we can rewrite (2.6) as

$$(\forall x \in D)(\text{Even}(x) \wedge x > 2) \Rightarrow P(x) \tag{2.7}$$

Note that we have increased the size of the domain. Thus, we may worry that our new statement (2.7) is not the same as the one we started with, so let's see that (2.6) and (2.7) are logically equivalent. Recall that an implication can only be false if the premise is true and the consequence is false. In our case, only if x is an even integer greater than 2 that does not satisfy P. We have increased the domain from D_1 to D, but the implication will hold for all x that are in D but not in D_1 by definition. Therefore, only an x from D_1 can invalidate the truth of (2.7), and such x would also invalidate the truth of (2.6). Conversely, an x that invalidates (2.6) would also invalidate (2.7), so we can conclude that (2.6) and (2.7) are logically equivalent.

Now we need to rewrite $P(x)$ which still has D_2 in it. We can write $P(x)$ as

$$(\exists y \in D_2) Q(x, y) \qquad \text{where} \quad Q(x, y) : (\exists z \in D_2) x = y + z$$

An integer y is prime if $\text{Prime}(y)$ is true. We are only interested in primes that add up to x, so we "and" that requirement with $Q(x, y)$. We do this instead of using an implication because if we used an implication, a number that is not prime would give us a false premise, thus making the implication true, which would make the existentially quantified statement true as well. But we only want primes to make $P(x)$ true, so we rewrite $P(x)$ using the domain D as

$$P(x) : (\exists y \in D)\text{Prime}(y) \wedge Q(x, y). \tag{2.8}$$

We would use the same strategy to eliminate the use of D_2 in the expression for $Q(x, y)$ to get

$$Q(x, y) : (\exists z \in D)\text{Prime}(z) \wedge x = y + z. \tag{2.9}$$

So now let's combine all that we just did to rewrite Goldbach's conjecture using a single domain $D = \{0, 1, 2, \ldots\}$. We start with (2.6):

$$(\forall x \in D_1)(\exists y \in D_2)(\exists z \in D_2) x = y + z.$$

Now apply (2.7) to get

$$(\forall x \in D)(\text{Even}(x) \wedge x > 2) \Rightarrow ((\exists y \in D_2)(\exists z \in D_2)x = y + z).$$

Now let's start getting rid of D_2, so we apply (2.8) and get

$$(\forall x \in D)(\text{Even}(x) \wedge x > 2) \Rightarrow ((\exists y \in D)\text{Prime}(y) \wedge ((\exists z \in D_2)x = y + z)).$$

Finally, apply (2.9) to get an expression that uses only a single domain.

$$(\forall x \in D)(\text{Even}(x) \wedge x > 2) \Rightarrow \big[(\exists y \in D)\text{Prime}(y) \wedge \big((\exists z \in D)\text{Prime}(z) \wedge x = y + z\big)\big] .$$