1. Propositional logic

Mathematics depends on precise statements for which the common (English) language is often too inaccurate and at times even plainly ambiguous. For this reason, mathematical statements are often cast in the dry language (and formalism) of propositional logic. We thus start by reviewing the important concepts from propositional logic that we will need throughout the rest of the course.

1.1 Logical propositions and operations.

A basic building block is a *proposition* which is simply a phrase or a statement (expressed using the *language* and *objects* of the theory) that can be assigned a TRUE/FALSE value. Examples of propositions is the statement $1 < 0$ or the statement

$$x^2 + 2x + 1 = 0 \text{ has no solution over } \mathbb{R}. \tag{1.1}$$

We will denote propositions P or Q in the sequel.

There are natural operations on propositions that produce new propositions. These correspond (roughly) to English words

$$\text{NOT, AND, OR, IMPLIES and IS EQUIVALENT TO} \tag{1.2}$$

For propositions P, Q these are denoted by

$$\neg P,\ P \wedge Q,\ P \vee Q,\ P \Rightarrow Q \text{ and } P \Leftrightarrow Q, \tag{1.3}$$

and are referred to as *negation*, logical *conjunction*, logical *disjunction*, *implication* and *equivalence*, respectively. To define these new propositions precisely, we give a list of their TRUE/FALSE values depending on those of P and Q. This results in

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE
TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE
FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE
FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	TRUE

which we refer to as the *truth table* for these operations.

Two points are worthy of a note. First, the OR represented by logical disjunction $\vee$ is not what is usually meant by "or" in idiomatic English. Indeed, when a host offers her guest a "beer or wine," she presumably means exactly one of the two while the logical OR — and likely also the thirsty guest — enjoys the possibility of taking both drinks concurrently. This arises because we often mistake OR for *exclusive-OR*, a.k.a. XOR, which is defined as

$$P \text{ XOR } Q := (P \wedge \neg Q) \vee (\neg P \wedge Q) \tag{1.4}$$

The logical OR is sometimes translated into English by the phrase and/or but that requires disambiguation with logical AND.

Another point to note is the somewhat unintuitive convention that if P is FALSE, then $P \Rightarrow Q$ is TRUE regardless of the truth value of Q. This subtle point shows the utmost importance of soundness of logical reasoning: Indeed, it implies that the introduction of

a single FALSE statement into the language or theory permits a logical inference of any statement and, consequently, the collapse of the whole theory.

Caution is at times needed to properly interpret a logical statement written in English as a statement in propositional logic. The most common situation arises with phrases "if" and "only if". These are meant to be read as follows

$$\text{“}P \text{ only if } Q\text{”} \text{ means } P \Rightarrow Q \tag{1.5}$$

while

$$\text{“}P \text{ if } Q\text{”} \text{ means } Q \Rightarrow P \tag{1.6}$$

One can easily err when converting the English phrase into the logical proposition so we should pay extra attention to these in what follows.

The above phrases commonly occur jointly in statements of form "P if and only if Q" which we take to mean $P \Leftrightarrow Q$. To give an example of the above, the phrase "x is positive only if x equals it absolute value" translates into

$$x > 0 \;\Rightarrow\; x = |x| \tag{1.7}$$

while "x is positive if x equals its absolute value" translates into

$$x = |x| \;\Rightarrow\; x > 0 \tag{1.8}$$

Note that the former statement is TRUE for all real-valued x, while the latter is FALSE for at least one x (namely, $x = 0$).

1.2 Primitive operations and equivalent forms.

Not all of the above operations are primitive; in fact, it would suffice to have just NOT and, for instance, OR. Indeed, with "$\neg$" and "$\vee$" defined by the corresponding columns in the above truth table, we could then set

$$\begin{aligned} P \wedge Q \;&:=\; \neg(\neg P \vee \neg Q) \\ P \Rightarrow Q \;&:=\; \neg P \vee Q \end{aligned} \tag{1.9}$$

and, with $\wedge$ and $\Rightarrow$ reduced to the primitives as above, let

$$P \Leftrightarrow Q \;:=\; (P \Rightarrow Q) \wedge (Q \Rightarrow P) \tag{1.10}$$

where we employ the convention that the NOT operation is always taken first, unless parentheses get in the way.

We also note that, here and henceforth,

$$\text{“}:=\text{” means “defined as”} \tag{1.11}$$

where the latter (unlike for ordinary equality) requires only that the object on the right-hand side is defined; the object on the left is then identified with that on the right (and so an ordinary equality henceforth applies).

That the first line (1.9) gives the desired concept is verified with the help of the fact that NOT is an *involution*,

$$\neg(\neg P) \;\Leftrightarrow\; P \tag{1.12}$$

and that the *de Morgan formulas*

$$\begin{aligned} \neg(P \wedge Q) \;&\Leftrightarrow\; (\neg P \vee \neg Q) \\ \neg(P \vee Q) \;&\Leftrightarrow\; (\neg P \wedge \neg Q) \end{aligned} \tag{1.13}$$

hold true. The second line in (1.9) is then deduced from the following lemma:

Lemma 1.1 (Proof by contradiction) *For any logical propositions P and Q,*

$$(P \Rightarrow Q) \;\Leftrightarrow\; \neg(P \wedge \neg Q) \tag{1.14}$$

Proof. This can be checked directly by working out the truth table of the proposition on the right-hand side. Alternatively, note that $P \Rightarrow Q$ is FALSE only if P is TRUE and Q is FALSE, which is also the only situation in which $P \wedge \neg Q$ is TRUE. □

The title of the lemma refers to the fact that (1.14) is the logical basis for proofs by contradiction. These demonstrate the implication "P implies Q" to be TRUE by way of assuming P along with the negation of Q and using these and a sequence of logical inferences to deduce an absurd (meaning: demonstrably FALSE) conclusion. That being said, the assumption that Q is FALSE is at times redundant in this argument and its omission leads to a somewhat different proof strategy:

Lemma 1.2 (Proof by contrapositive) *For any logical propositions P and Q,*

$$(P \Rightarrow Q) \;\Leftrightarrow\; (\neg Q \Rightarrow \neg P) \tag{1.15}$$

Proof. We again readily check that both sides are FALSE only for the situation in which P is TRUE and Q is FALSE. □

While the unary operation NOT must be included among the primitives, the reliance on OR as the primitive binary operation is a matter of choice. Indeed, by de Morgan formulas, AND could be used instead and, in light of (1.9), IMPLIES could be used as well. Note that although 16 distinct binary operations exist on propositions P and Q, they can all be written as strings of logical disjunctions of logical conjunctions involving P, $\neg P$, Q and $\neg Q$ and are thus reducible to NOT and OR above. That being said, the reduction to primitives is important mainly for theory building, and perhaps implementation on a computer, so we will keep using the full set above in the sequel.

1.3 Predicates and quatifiers.

Most of mathematics deals with propositions that depend on parameters. Such propositions are called *predicates* or *Boolean functions*. The TRUE/FALSE value of the predicate $P(x)$ is generally not determined until x is specified. In order to turn predicates into proper propositions, we thus need to specify, or curb, x somewhat. This is done using the following *quantifiers*

$$\begin{aligned} \forall &:= \text{ for all} \\ \exists &:= \text{ there exists} \end{aligned} \tag{1.16}$$

With the help of these we define propositions

$$\forall x\colon P(x) \;:=\; P(x) \text{ is TRUE for every choice of } x \tag{1.17}$$

where, alternatively, the phrase on the right means that the Boolean function $x \mapsto P(x)$ has no FALSE in its range. Similarly,

$$\exists x\colon P(x) \;:=\; P(x) \text{ is TRUE for at least one choice of } x \tag{1.18}$$

which in the language of Boolean functions means $x \mapsto P(x)$ has TRUE in its range. Another word used in the above context is *satisfiability*. The phrase $\exists x\colon P(x)$ means that $P(x)$ is satisfiable while $\forall x\colon P(x)$ means that $\neg P(x)$ is NOT satisfiable.

We may often need to further restrict the x for which the quantifier applies. This requires the language of set theory which, for the sake of easier explanation, we assume the reader to be familiar with. Given a set E, we write

$$\forall x \in E\colon P(x) \;:=\; \forall x\colon\; x \in E \Rightarrow P(x) \tag{1.19}$$

and

$$\exists x \in E\colon P(x) \;:=\; \exists x\colon\; x \in E \wedge P(x) \tag{1.20}$$

Examples of these are the predicates (depending on a set $E \subseteq \mathbb{R}$)

$$\forall x \in E\colon x^2 > 0 \tag{1.21}$$

which is TRUE if $0 \notin E$ and FALSE otherwise and

$$\exists x \in E\colon x^2 > 0 \tag{1.22}$$

which is FALSE if $E \subseteq \{0\}$ and TRUE otherwise.

Having both quantifiers around is again done mainly for comfort of expression as one can be reduced to the other. Indeed, we have:

Lemma 1.3 *For any predicate $P(x)$,*

$$\begin{aligned} \neg\big(\forall x\colon P(x)\big) &\Leftrightarrow \exists x\colon \neg P(x) \\ \neg\big(\exists x\colon P(x)\big) &\Leftrightarrow \forall x\colon \neg P(x) \end{aligned} \tag{1.23}$$

Proof. We will only prove the first equivalence employing the formalism of Boolean functions. Indeed, $\forall x\colon P(x)$ means that the range of the map $x \mapsto P(x)$ is the singleton $\{\text{TRUE}\}$. Since $P(x)$ takes only TRUE/FALSE values, the negation of $\forall x\colon P(x)$ means that the range of $x \mapsto P(x)$ contains FALSE. But this is the same as saying that the range of $x \mapsto \neg P(x)$ contains TRUE, which is then identified with $\exists x\colon \neg P(x)$. □

Formulas can depend on more than just one parameter some of which may need to be quantified. We use the convention that while quantifiers are read left to right, they are applied right to left. Indeed, given a formula $P(x,y)$ in two variables, we have

$$\forall x\, \exists y\colon P(x,y) \;:=\; \forall x\colon \Big(\exists y\colon P(x,y)\Big) \tag{1.24}$$

and, similarly,

$$\forall x\, \forall y\colon P(x,y) \;:=\; \forall x\colon \Big(\forall y\colon P(x,y)\Big) \tag{1.25}$$

As with most other operations in mathematics, the order quantifiers are applied matters for the result. We in fact have:

Lemma 1.4 *For any predicate $P(x,y)$,*

$$\forall x \forall y\colon\; P(x,y) \;\Leftrightarrow\; \forall y \forall x\colon\; P(x,y) \tag{1.26}$$

and

$$\exists x \exists y\colon\; P(x,y) \;\Leftrightarrow\; \exists y \exists x\colon\; P(x,y) \tag{1.27}$$

and

$$\exists x \forall y\colon\ P(x,y) \ \Rightarrow\ \forall y \exists x\colon\ P(x,y) \tag{1.28}$$

We leave the easy proof (based on similar arguments as those in the proof of Lemma 1.3) to the reader. Note that for mixed quantifiers, only one implication is claimed above because the other does not hold in general. For instance,

$$\forall m \in \mathbb{N}\, \exists n \in \mathbb{N}\colon\ m = n \tag{1.29}$$

is a TRUE statement while the one with the quantifiers reversed is not.

Parentheses have to be used whenever possible ambiguity arises. The position of the parentheses can of course change the logical content of the statement. For instance,

$$\forall n \in \mathbb{N}\colon \Big(\exists m \in \mathbb{N}\colon m = n \Rightarrow 0 = 1\Big) \tag{1.30}$$

is a TRUE statement yet

$$\Big(\forall n \in \mathbb{N}\, \exists m \in \mathbb{N}\colon m = n\Big) \Rightarrow 0 = 1 \tag{1.31}$$

is FALSE. (Here $\mathbb{N}$ are the naturals with the usual interpretation of 0 and 1.)