Learning Objectives

By the end of this worksheet, you will:

- Translate sentences between natural English and predicate logic.

- Use definitions of predicates to simplify or expand statements in predicate logic.

- Apply negation equivalence rules to simplify statements in predicate logic.

1. **Translation with predicates**. Suppose we have a set P of computer programs that are each meant to solve the same task. Some of the programs are written in the Python programming language, and some are written in a different language. Some of the programs correctly solve the task, and others do not.

   Let's define the following predicates:

   $$Python(x) : \text{``}x \text{ is a Python program''},   \quad \text{where } x \in P$$
   $$Correct(x) : \text{``}x \text{ solves the task correctly''},   \quad \text{where } x \in P$$

   Express each English statement below as a sentence in predicate logic.

   (a) Program my_prog is correct and is written in Python.

   > **Solution**
   >
   > $Correct(my\_prog) \wedge Python(my\_prog)$

   (b) An incorrect program is written in Python.

   > **Solution**
   >
   > $\exists x \in P,\ \neg Correct(x) \wedge Python(x)$

   (c) No Python program is correct.

   > **Solution**
   >
   > $\neg\,(\exists x \in P,\ Python(x) \wedge Correct(x))$
   > or
   > $\forall x \in P,\ Python(x) \Rightarrow \neg Correct(x)$

   (d) Every incorrect program is written in Python.

   > **Solution**
   >
   > $\forall x \in P,\ \neg Correct(x) \Rightarrow Python(x)$
   > or the contrapositive,
   > $\forall x \in P,\ \neg Python(x) \Rightarrow Correct(x)$

   Translate each predicate logic statement below into natural English.

   (a) $\exists x \in P,\ Python(x) \wedge Correct(x)$

   > **Solution**
   >
   > At least one Python program is correct.

   (b) $\forall x \in P,\ \neg Python(x) \wedge Correct(x)$

> **Solution**
>
> Every program is written in a language other than Python and is correct.

(c) $\neg\big(\forall x \in P,\ Correct(x) \Rightarrow Python(x)\big)$

> **Solution**
>
> Not every correct program is written in Python.
> *Alternate*: A correct program is written in a language other than Python.

(d) $\forall x \in P,\ \neg Python(x) \Leftrightarrow Correct(x)$

> **Solution**
>
> For every program x, x is not written in Python if and only if x is correct.
> But it's a bit more natural to first rewrite the $\Leftrightarrow$ and then translate:
>
> - $\forall x \in P,\ \big(\neg Python(x) \Rightarrow Correct(x)\big) \wedge \big(Correct(x)\big) \Rightarrow \neg Python(x)\big)$, which translates to "Every program not written in Python is correct, and every correct program is not written in Python."
> - $\forall x \in P,\ \big(\neg Python(x) \wedge Correct(x)\big) \vee \big(Python(x) \wedge \neg Correct(x)\big)$, which translates to "Every program is either not written in Python and correct, or written in Python and not correct."

2. **Quantifiers in subformulas**. So far, we have seen quantifiers only as the leftmost components of our formulas. However, because all predicate statements have truth values (i.e., are either True or False), they too can be combined using the standard propositional operators. Let's see some examples of this (refer to the same predicates as Question 1).

   (a) Translate the following statement into English.

   $$\Big(\forall x \in P,\ Python(x) \Rightarrow Correct(x)\Big) \vee \Big(\forall y \in P,\ Python(y) \Rightarrow \neg Correct(y)\Big)$$

   > **Solution**
   >
   > All Python programs are correct, or all Python programs are incorrect.

   (b) Translate the following statement into predicate logic. "If at least one Python program is correct, then all Python programs are correct."

   > **Solution**
   >
   > $\big(\exists x \in P,\ Python(x) \wedge Correct(x)\big) \Rightarrow \big(\forall y \in P,\ Python(y) \Rightarrow Correct(y)\big)$

   (c) Finally, consider the following two statements:

   $$\big(\exists x_1 \in \mathbb{N},\ x_1 \mid 165\big) \wedge \big(\exists x_2 \in \mathbb{N},\ 7 \mid x_2\big) \qquad\qquad \text{(Statement 1)}$$
   $$\exists x \in \mathbb{N},\ x \mid 165 \wedge 7 \mid x \qquad\qquad \text{(Statement 2)}$$

   What is the difference between these two statements? Are they True or False?

   > **Solution**
   >
   > In the first statement, there are two different numbers, and each one satisfies one predicate (dividing 165 or being divisible by 7). In the second statement, we only have one number, that must satisfy both predicates.
   > The first statement is True (why?) and the second statement is False (why?).

3. **Expanding definitions**. Consider the following statement:

If m and n are odd integers, then mn is an odd integer.

If we want to express this statement using predicate logic, we need to start with a definition of the term "odd". Let $n \in \mathbb{Z}$. We say that n is **odd** when $2 \mid (n+1)$, i.e., when $\exists k \in \mathbb{Z}, n+1 = 2k$.

(a) Write the definition of a predicate over the integers named Odd that is True when its argument is odd.

> **<u>Solution</u>**
>
> $$Odd(x): \ \exists k \in \mathbb{Z}, x+1 = 2k, \qquad \text{where } x \in \mathbb{Z}.$$
>
> Or
>
> $$Odd(x): \ 2 \mid (x+1), \qquad \text{where } x \in \mathbb{Z}.$$

(b) Using the predicate Odd and the notation of predicate logic, express the statement:

> For every pair of odd integers m and n, mn is an odd integer.

> **<u>Solution</u>**
>
> $\forall m, n \in \mathbb{Z}, Odd(m) \wedge Odd(n) \Rightarrow Odd(mn)$

(c) Repeat part (b) but fully expand the definitions of the predicates Odd and $\mid$.

> **<u>Solution</u>**
>
> $$\forall m, n \in \mathbb{Z}, \ \Big(\big(\exists k_1 \in \mathbb{Z}, \ m = 2k_1 + 1 \big) \wedge \big(\exists k_2 \in \mathbb{Z}, \ n = 2k_2 + 1 \big) \Big) \Rightarrow \big(\exists k_3 \in \mathbb{Z}, \ mn = 2k_3 + 1 \big)$$

(d) Repeat parts (b) and (c) using the following statement (which states the converse of the original implication).

> For every pair of integers m and n, if mn is odd, then m and n are odd.

4. **Simplifying negated formulas**. Recall the rules governing how to simplify negations of predicate formulas:

- $\neg(\neg p)$ becomes p.
- $\neg(p \vee q)$ becomes $\neg p \wedge \neg q$.
- $\neg(p \wedge q)$ becomes $\neg p \vee \neg q$.
- $\neg(p \Rightarrow q)$ becomes $p \wedge \neg q$.
- $\neg(p \Leftrightarrow q)$ becomes $(p \wedge \neg q) \vee (\neg p \wedge q)$.
- $\neg(\exists x \in S, \ P(x))$ becomes $\forall x \in S, \ \neg P(x)$.
- $\neg(\forall x \in S, \ P(x))$ becomes $\exists x \in S, \ \neg P(x)$.

Using these rules, simplify each of the following formulas so that the negations are applied directly to predicates/propositional variables. Note: this is a pretty mechanical exercise, but an extremely valuable one: once we get to the next chapter, we will be assuming you can take negations of statements very quickly as a first step in some proofs.

(a) $\neg\Big((a \wedge b) \Leftrightarrow c \Big)$

> **<u>Solution</u>**

$$\neg\Big((a \wedge b) \Leftrightarrow c\Big)$$

$$\Big((a \wedge b) \wedge (\neg c)\Big) \vee \Big((\neg(a \wedge b) \wedge c)\Big)$$

$$\Big(a \wedge b \wedge \neg c\Big) \vee \Big((\neg a \vee \neg b) \wedge c\Big)$$

(b) $\neg\Big(\forall x, y \in S, \ \exists z \in S, \ P(x,y) \wedge Q(x,z)\Big)$

Solution

$$\neg\Big(\forall x, y \in S, \ \exists z \in S, \ P(x,y) \wedge Q(x,z)\Big)$$

$$\exists x, y \in S, \ \forall z \in S, \ \neg\Big(P(x,y) \wedge Q(x,z)\Big)$$

$$\exists x, y \in S, \ \forall z \in S, \ \neg P(x,y) \vee \neg Q(x,z)$$

(c) $\neg\Big((\exists x \in S, \ P(x)) \Rightarrow (\exists y \in S, \ Q(y))\Big)$

Solution

$$\neg\Big((\exists x \in S, \ P(x)) \Rightarrow (\exists y \in S, \ Q(y))\Big)$$

$$(\exists x \in S, \ P(x)) \wedge \neg(\exists y \in S, Q(y))$$

$$(\exists x \in S, \ P(x)) \wedge (\forall y \in S, \neg Q(y))$$

5. **Choosing a universe and predicates**. Consider the statement

$$\Big[(\exists x \in U, \ P(x)) \wedge (\exists y \in U, \ Q(y))\Big] \Rightarrow \Big[\exists z \in U, \ P(z) \wedge Q(z)\Big].$$

Define a non-empty domain U and predicates P and Q for which this statement is False.

Hint: The statement says: "If some $x \in U$ makes $P(x)$ True and some $y \in U$ makes $Q(y)$ True, then some $z \in U$ makes both $P(z)$ and $Q(z)$ True." Think about how this statement could be False, and use this to construct a U, P and Q.

Solution

Since we want to show that an implication is False, we want to construct an example where the hypothesis is True and the conclusion False.

Let $U = \mathbb{Z}$. Define the following predicates over $\mathbb{Z}$: $P(x) :$ "x is even", and $Q(x) :$ "x is odd".

Then $(\exists x \in U, \ P(x))$ is True, as the integer 2 provides an example. And $(\exists y \in U, \ Q(y))$ is also True, as the integer 3 provides an example. The hypothesis in the statement is therefore True.

However, there is no integer that is both even and odd at the same time, and so $[\exists \ z \in U, \ P(z) \wedge Q(z)]$ is False. (Put another way, the intersection of the even and odd numbers is empty.) So the conclusion in the statement is False.