

syn.1 Introduction

sol:syn:int:
sec

In first-order logic, we combine the non-logical symbols of a given language, i.e., its **constant symbols**, **function symbols**, and **predicate symbols**, with the logical symbols to express things about first-order **structures**. This is done using the notion of satisfaction, which relates a **structure** \mathfrak{M} , together with a variable assignment s , and a **formula** φ : $\mathfrak{M}, s \models \varphi$ holds iff what φ expresses when its **constant symbols**, **function symbols**, and **predicate symbols** are interpreted as \mathfrak{M} says, and its free variables are interpreted as s says, is true. The interpretation of the **identity predicate** $=$ is built into the definition of $\mathfrak{M}, s \models \varphi$, as is the interpretation of \forall and \exists . The former is always interpreted as the identity relation on the **domain** $|\mathfrak{M}|$ of the structure, and the quantifiers are always interpreted as ranging over the entire **domain**. But, crucially, quantification is only allowed over elements of the **domain**, and so only **object variables** are allowed to follow a quantifier.

In second-order logic, both the language and the definition of satisfaction are extended to include free and bound function and predicate variables, and quantification over them. These variables are related to **function symbols** and **predicate symbols** the same way that object variables are related to **constant symbols**. They play the same role in the formation of terms and **formulas** of second-order logic, and quantification over them is handled in a similar way. In the *standard* semantics, the second-order quantifiers range over all possible objects of the right type (n -place functions from $|\mathfrak{M}|$ to $|\mathfrak{M}|$ for function variables, n -place relations for predicate variables). For instance, while $\forall v_0 (P_0^1(v_0) \vee \neg P_0^1(v_0))$ is a formula in both first- and second-order logic, in the latter we can also consider $\forall V_0^1 \forall v_0 (V_0^1(v_0) \vee \neg V_0^1(v_0))$ and $\exists V_0^1 \forall v_0 (V_0^1(v_0) \vee \neg V_0^1(v_0))$. Since these contain no free variables, they are **sentences** of second-order logic. Here, V_0^1 is a second-order 1-place predicate variable. The allowable interpretations of V_0^1 are the same that we can assign to a 1-place **predicate symbol** like P_0^1 , i.e., subsets of $|\mathfrak{M}|$. Quantification over them then amounts to saying that $\forall v_0 (V_0^1(v_0) \vee \neg V_0^1(v_0))$ holds for all ways of assigning a subset of $|\mathfrak{M}|$ as the value of V_0^1 , or for at least one. Since every set either contains or fails to contain a given object, both are true in any **structure**.

Photo Credits

Bibliography