Specifying Systems Notes

by Arnav Kumar

Web: https://arnavcs.github.io

Specifying Systems is a publication written by Leslie Lamport on the TLA$^+$ language. This is a collection of notes that I make about the text as I read it, and is not a summary or recreation of the text, but rather a reference for anyone who has already read the text. As such, please read the text to gain a better understading of the contents.

1 System Specifications

Color Scheme Key
Definition
Note

System Specification

A system specification is a description of what a system should do or is intended to do. The behavioural properties of a system are also called the functional or logical properties of a system and is our focus. We do not consider performance properties.

Temporal Logic

A temporal logic formula is a formula that describes a system's behaviour by relating the next state of a system with the current state.

TLA$^+$

TLA$^+$ stands for the Temporal Logic of Actions and supports both assertional resoning along with temporal logic. This system is quite good with describing asynchronous systems, but can be used for nearly any purpose: APIs and distributed systems included.

Propositional Logic

The two basic boolean values, TRUE and FALSE can be used in propositional logic with the operators $\neg$, $\wedge$, $\vee$, $\Rightarrow$, and $\equiv$ (from highest to lowest precendence).

Tautology

A tautology is a proposition that is true for all possible truth values of its identifiers. For example, the following logic-proposition is a tautology:

$$F \Rightarrow F \vee G$$

Sets

A set is a collection of elements that is determined by its elements. We denote sets with curly brackets, so the set of the first three natural numbers is $\{1, 2, 3\}$. The empty set will be denoted as $\{\}$, and operations on sets are $\cap$, $\cup$, $\subseteq$, and $\setminus$. Membership is denoted with $\in$.

Predicate Logic

The two quantifiers, $\forall$ and $\exists$, are followed with a colon and the variable in the quantifier is called "bound" as opposed to a "free" variable. See the example below where x and y are both bound.

$$\forall x \in S : (\forall y \in T : F)$$

Formulas vs. Statements

Note that by default, something like $2 * x > x$ is a noun; it is true or false depending on the value of x. On the other hand, if we would like to assert if the formula is true, then we should instead write the statement $2 * x > x$ is true.