

Основы логики излагаются в стиле, близком к учебному пособию [12], особенно в части формализмов логических языков и исчисления предикатов первого порядка, а также его расширений. Понятие многосортного индуктивного класса является распространением на многосортный случай классического понятия индуктивного класса, взятого из [11]. Из других работ по математической логике, использованных в изложении данной главы, отметим [32]. Пример расширения арифметики целых чисел, на операции над массивами есть адаптация примера из [59]. Изложение принципов логической спецификации программ с использованием инвариантов основывается на [27, 44, 45].

Логический язык спецификаций свойств программ по своим идеям ближе всего к языку Jota [17], однако использует более широкие средства композиции термов, основанные на обобщенном подходе к квантификации, аналогичном концепции кванторов языка SETL и иерархическом представлении сложных термов по принципу «сверху вниз». Основы методики логической спецификации программ опираются на средства структуризации языка логической спецификации свойств программ и базируются на исследованиях автора по верификации программ на подмножестве языка ПЛ/1.

Многие примеры взяты из [18, 59].

ГЛАВА 3.

АНАЛИЗ КОРРЕКТНОСТИ ПОСЛЕДОВАТЕЛЬНЫХ ПРОГРАММ

3.1. ОСНОВНЫЕ ЗАДАЧИ АНАЛИЗА КОРРЕКТНОСТИ ПРОГРАММ

Свойство программы, характеризующее отсутствие ошибок в программе по отношению к целям разработки, назовем *корректностью* (правильностью) *программы*. Корректность программы есть статическое свойство, в отличие от классического понятия безотказности, используемого в теории надежности технических систем. Оно не зависит от времени (если, конечно, цели разработки неизменны!), что отражает специфику ошибок разработки (ошибок проекта и кодирования) по сравнению с ошибками износа, характерными для работы аппаратуры.

Корректность программы — относительное понятие, оно имеет смысл только для фиксированной (и точно определенной!) цели разработки программы. Основная задача верификации состоит в демонстрации свойства корректности программы. Более точно ее следует определить так: *верификация* — установление соответствия