

Аналогичным образом могут быть формализованы и другие перечисленные выше свойства корректности, являющиеся отрицанием свойств завершения, частичной корректности и тотальной корректности.

Полученные формализации свойств корректности программ создают основу для формального логического анализа корректности, т. е. доказательства присутствия соответствующих свойств в рамках некоторых формальных (аксиоматических) систем. Прежде чем переходить к изложению соответствующих методик логического анализа свойств корректности (методов верификации), отметим, что разделение свойств частичной корректности и завершения следует рассматривать как методологический прием, направленный на уменьшение сложности соответствующих доказательств.

В общей постановке (для произвольных программ) каждое из этих свойств (завершение и частичная корректность) является алгоритмически неразрешимым, т. е. не существует алгоритма, позволяющего оценить истинность произвольной тройки Хоара $\{P\}Prgm\{Q\}$ или предиката $fin(x)$. Однако для различных классов программ могут быть развиты достаточно эффективные методы анализа свойств корректности, позволяющие вручную или в автоматизированном режиме проводить соответствующие доказательства. Такие методы могут рассматриваться как составная часть процессов разработки и отладки программ, обеспечивая логическую завершенность этих процессов.

3.2. МЕТОД ИНДУКТИВНЫХ УТВЕРЖДЕНИЙ

Метод индуктивных утверждений — наиболее распространенный метод доказательства частичной корректности программ. Он позволяет свести доказательство свойства частичной корректности программы к доказательству некоторого (конечного) числа утверждений, записанных в виде формул логического языка спецификации и имеющих интерпретацию в соответствующей проблемной области. Этот метод составляет основу большинства разработанных автоматизированных систем верификации.

Рассмотрим метод индуктивных утверждений для программ с конечным множеством простых переменных $\{x_1, x_2, \dots, x_n\}$, где $n \geq 1$, а операторами являются операторы присваивания вида $x_i := f(x_1, \dots, x_i, \dots, x_n)$; составной оператор вида $A; B$, где A, B — операторы; оператор условного перехода

$\text{if } \alpha(x_1, \dots, x_i, \dots, x_n) \text{ then } L_+ \text{ else } L_-$,

где L_+, L_- — метки операторов, которым передается управление.

Начальным и конечным операторами являются соответственно START и STOP. Такой состав структурных компонентов вполне