

$f:d(t_1, \dots, t_i, \dots, t_n)$ или $f(x_1, \dots, x_m):d(t_1, \dots, t_i, \dots, t_n)$,

где f — имя определяемой функции; d, t_1, \dots, t_n — термы; x_i — переменные, от которых зависит f .

Иерархическая структура формул есть последовательность определений функций. Она позволяет рассматривать сложные формулы в соответствии со стратегией «сверху вниз», абстрагируясь на каждом уровне иерархии от несущественных для него деталей представления, которые можно восстановить на других уровнях.

Пример 2.6. Иерархическая структура формул $\varphi_1 \div \varphi_3$ представлена в виде

$\varphi_1 : d_1(x_1, x_2);$

$\varphi_2 : d_2(t_1, t_2);$

$\varphi_3 : d_3(t_1, t_3);$

$t_1 : d_4(y_1);$

$t_3 : d_5(y_2).$

Формулы иерархической структуры могут быть преобразованы в обычные формулы подстановкой «снизу вверх», например

$\varphi_3 : d_3(d_4(y_1), d_5(y_2)).$

2.3.2. Аксиоматические теории проблемных областей

Как мы уже отмечали, интерпретация языка практически необходима для того, чтобы наделить формулы языка LS соответствующей семантикой, т. е. связать элементы формул с объектами и отношениями рассматриваемой области. В нашем случае (применительно к спецификации свойств программ) интерпретация включает:

задание используемых типов данных, сигнатуры и схем функций и предикатов, используемых в спецификациях;

задание семантики функций и предикатов.

Для аксиоматических теорий семантика задается системой аксиом-формул, истинных в данной проблемной области и расширяющих набор логических аксиом базового исчисления предикатов. Таким образом, аксиоматическая теория рассматриваемой области позволяет устанавливать истинность формул языка путем вывода этих формул в данной теории.

Описание аксиоматической теории на языке LS условимся разделять на синтаксическую и семантическую части. Синтаксическая часть, называемая иногда интерфейсной, определяет множество используемых типов, сигнатуру и схемы функций и предикатов. Множество используемых типов выбирается среди типов данных базового языка программирования, кроме того, новые типы могут конструироваться средствами языка LS как абстрактные типы данных. Для описания сигнатуры предоставляются широкие воз-