

Процик Петро Павлович,
Кафедра теорії та технології програмування,
Київський національний університет ім. Тараса Шевченка,
01033, Київ, вул. Володимирська, 69,
E-mail: pprocik@unicyb.kiev.ua

Композиційно-номінативна семантика мови специфікації Z-notation

Z-Notation - формальна мова специфікації програмних систем. Вона використовується для опису, моделювання та доведення властивостей функціональних аспектів послідовних систем.

Специфікація програмної системи у Z-Notation будується за допомогою послідовності схем. Схема складається з двох частин – декларативної та предикативної. У декларативній частині проводиться опис змінних, а у предикативній на змінні накладаються обмеження. Існує три типи змінних – вхідні, вихідні та змінні стану системи. Синтаксично схема має такий вигляд:

[Declaration | Predicates].

Існує декілька підходів до побудови семантики Z схем. Один з них полягає у тлумаченні схеми, як множини усіх можливих підстановок значень у змінні, які описані у декларативній частині та задовольняють накладені обмеження [1]. Можна розглядати схему, як цілісну сутність, та будувати на цій підставі числення та логічні системи [2].

У доповіді пропонується застосування композиційно-номінативного підходу [3,4] для побудови семантики мови схем. Використання підходу дозволяє будувати семантику формальної мови на підставі єдиної методологічної платформи. Згідно принципу розвитку спочатку розглядається найбільш абстрактне тлумачення схем, а далі, шляхом поступового уточнення, відбувається перехід до конкретної мови. На кожному рівні будується відповідна семантика.

Стани системи подаються за допомогою поняття номінативної множини. Таким чином стан складається з компонентів вигляду ім'я→значення. Переходи тлумачаться, як номінативні операції, які відображають одні стани системи у інші.

На першому кроці будується семантика спрощеної мови AZ (Abstract Z). На відміну від мови Z, AZ – буде без типовою мовою. Усі змінні приймають значення з деякого загального універсуму значень. Операції діють над усім станом системи. Вхідні та вихідні значення, а також локальні змінні відсутні.

Далі, на основі введених семантичних визначень можна побудувати транзиційну модель системи, що задається відповідною специфікацією. Існує можливість перевірки властивостей побудованої транзиційної системи, а це, як наслідок, відкриває один з шляхів дослідження властивостей вхідної специфікації.

Навіть на такому, досить високому, рівні абстракції можна будувати специфікації та моделі різних типів систем. Це означає, що рівень є прагматично корисним і його дослідження має практичний інтерес.

Подальший розвиток семантики полягає у дослідженні властивостей предикатів, що задають операції. Розгляді розширення класів змінних, введення поняття локальних визначень та дослідженні семантичних зв'язків між різними рівнями.

ЛІТЕРАТУРА:

1. Jim Woodcock, Jim Davies : Prentice Hall - Using Z. Specification, Refinement, and Proof, 523 p, 1996.
2. Brien S.M., Martin A.P. : Academic Press – Symbolic Computation volume 11 : A Calculus for Schemas in Z, pages 1-29, 1999.
3. Никитченко Н.С. Композиционно-номинативный подход к уточнению понятия программы : Проблемы программирования.-1999.-№1, стр. 16-31.

4. Редько В.Н. Основания композиционного программирования: Программирование. – 1973.-№3-стр. 3-13.

DRAFT