Дедуктивная верификация программ на языке Python

Бикбулатов Тимур Русланович

Студент

Факультет ВМК МГУ имени М. В. Ломоносова, Москва, Россия

E-mail: bikbulatovtimur96@yandex.ru

Научный руководитель *— Корухова Юлия Станиславовна*

Возрастающая сложность программного обеспечения становится причиной большого количества ошибок в нём, а одновременный рост критичности выполняемых им функций влечёт ущерб от этих ошибок. Для обеспечения надежности большое значение имеет верификация, выявляющая ошибки на этапе разработки. Она проверяет соответствие программы заранее заданным требованиям, сформулированным в виде набора логических условий. При верификации исходного кода анализируются следующие характеристики [2]:

- Код написан в соответствии с синтаксическими и семантическими правилами выбранных языков программирования.

- В исходном коде отсутствуют пути выполнения, достижимые в условиях работы программы и приводящие к её сбоям.

Одним из методов верификации является дедуктивный анализ. Проверка программы организована следующим образом [1]:

1. Спецификация программы в виде её предусловия и постусловия определяется в рамках исчисления высказываний.

2. В коде программы выбираются точки сечения так, чтобы любой цикл содержал по крайней мере одну такую точку.

3. Для каждой точки находится предикат, характеризующий отношения между переменными. В начале программы в качестве предиката выбирается предусловие, в конце — постусловие.

4. Программа разбивается на набор возможных линейных путей между парами точек сечения. Для каждого такого пути P_{ij} между точками i и j проверяется истинность тройки $iP_{ij}j$, обозначающая, что после выполнения P будет истинным.

Существуют различные инструменты верифицирования программ методом дедуктивного анализа. Преимущественно все верификаторы работают с языком Си. В рамках работы сделан сравнительный анализ таких систем. Эксперименты показали, что перевод программы с Python на Си на выходе выдаёт текст программы во много раз превышающий размер изначального, поэтому такие инструменты не подошли.

Был реализован алгоритм верификации программ на языке Python методом дедуктивного анализа. Данный язык программирования является динамическим и нетипизированным, потому существуют сложности, затрудняющие полноценную проверку кода. Было принято решение взять подмножество синтаксиса Python, позволяющее применить верификацию. В качестве инструмента выбрана платформа Why3 с использованием доказывателей Alt-Ergo, CVC4, Z3 [3]. Для взаимодействия с данной платформой был создан модуль перевода кода с Python в код на языке внутреннего представления WhyML платформы Why3.

В результате была создан прототип системы, позволяющий выполнить проверку программы на языке Python методом дедуктивного анализа.

Литература

1. Камкин А. С. Введение в формальные методы верификации программ: учебное пособие. М.: МАКС Пресс, 2018.
2. Fitting M. First-Order Logic and Automated Theorem Proving // Graduate Texts in Computer Science, New York, 1990, pp. 45-50.
3. Описание работы платформы Why3: `http://why3.lri.fr/ssft-16/notes-why3.pdf`