

Лабораторная работа № 1.

Случайное тестирование на основе свойств. Спецификация основных категорных понятий

1. Цель работы

Изучение подхода и систем случайного тестирования на основе свойств. Изучение основных конструкций теории категорий и выявление их аналогов в системах программирования. Построение и проверка спецификаций для найденных аналогов.

2. Программное обеспечение

Для выполнения лабораторных работ по курсу следует выбрать язык программирования и библиотеку случайного тестирования на основе свойств для этого языка.

3. Теоретические сведения

Случайное тестирование на основе свойств (СТОС) — это механизм построения спецификаций программ в виде логических высказываний, введенный в системе QUICKCHECK [1] для языка программирования HASKELL. В настоящее время создано множество аналогов QUICKCHECK для всех популярных языков программирования (ЯП).

На практике подход СТОС применяется для верификации программ и программных систем с использованием не отдельных тестов, проверяющих поведение программы в конкретных случаях, а, вместо этого, более общих свойств, покрывающих отдельные области поведения программы множеством тестов, использующих случайные значения.

На формальном уровне, близость структуры свойств синтаксису логики первого порядка, а точнее выражениям с кванторами всеобщности, дает возможность понимать спецификации программ в виде набора свойств как логические спецификации программы, описывающие ее поведение как математического объекта.

Для рассмотрения программ как математических конструкций уместен подход функционального программирования, аппликативных вычислительных систем [2] и теории категорий [3]. Парадигма функционального программирования дает общий взгляд на программы как на чистые математические функции. Аппликативные вычислительные системы (АВС) предоставляют систему вычислений, основывающуюся исключительно на функциях и применении («аппликации») функций друг к другу.

Теория категорий исследует общие свойства математических теорий путем сведения их к коллекциям «стрелок», или «морфизмов». Помимо морфизмов обычно вводится понятие «объектов». В рамках категории множеств морфизмы воспринимаются как функции, а объекты как множества, которые могут выступать в роли области определения («домена», dom) или области значений («кодомена», cod) морфизмов. Выделяются т. н. Hom -множества, содержащие морфизмы между заданными объектами:

$$\text{Hom}(A, B : \text{Set}) := \{ f \mid \text{dom}(f) = A \wedge \text{cod}(f) = B \}.$$

Будем считать, что записи $f : A \rightarrow B$ и $f \in \text{Hom}(A, B)$ имеют одинаковое значение.

4. Задания

4.1. Выбор платформы

Выбрать язык программирования и реализацию системы случайного тестирования на основе свойств для этого языка программирования. Кратко описать составные части системы: представление свойств, методов генерации значений, способов запуска тестов для проверки свойств.

С целью изучения способов работы со свойствами сформулировать и проверить спецификацию функции сортировки списков.

4.2. Основные конструкции

Найти, каким образом в выбранном ЯП представляются функции.

Сформулировать свойство экстенционального равенства двух функций $f, g \in \text{Hom}(A, B)$:

$$f =_{\text{Hom}(A, B)} g := \forall x, y \in A (f(x) = g(x))^1.$$

Есть ли в выбранной библиотеке СТОС возможность абстрагировать данную конструкцию до «производного» свойства — т. е. функции

$$\text{eq}(f, g \in \text{Hom}(A, B)) := \forall x, y \in A (f(x) = g(x))?$$

4.3. Свойства композиции

На функциях определить операцию композиции: для функций $f : B \rightarrow C$ и $g : A \rightarrow B$

$$(f \circ g)(x) = f(g(x)).$$

Сформулировать и проверить свойство ассоциативности композиции:

$$\begin{aligned} \text{compAss}(A, B, C, D : \text{Set}) &:= \\ \forall f \in \text{Hom}(C, D), g \in \text{Hom}(B, C), h \in \text{Hom}(A, B) & \\ (f \circ g) \circ h =_{\text{Hom}(A, D)} f \circ (g \circ h). & \end{aligned}$$

Приведенное выше определение параметризовано множествами A, B, C, D . В большинстве реализаций СТОС такая параметризация соответствовала бы полиморфным свойствам, параметризованным типами, которые обычно не поддерживаются. Поэтому в рамках выполнения лабораторной работы следует приводить примеры подобных свойств для конкретных типов (т. е. множеств генерируемых для тестов значений).

4.4. Идентичность

Определить функцию идентичности $\text{id}_A(x : A) := x$ и проверить свойства идентичности:

$$\begin{aligned} \text{idL}(A, B : \text{Set}) \forall &:= f \in \text{Hom}(A, B) (\text{id}_B \circ f = f), \\ \text{idR}(A, B : \text{Set}) \forall &:= f \in \text{Hom}(A, B) (f \circ \text{id}_A = f). \end{aligned}$$

¹Заметим, что равенство, и, в частности экстенциональное, определяется в контексте множества сравниваемых предметов, в указанном определении это множество $\text{Hom}(A, B)$.

4.5. Терминальный объект

Терминальный объект будем обозначать как 1, он оснащается следующей спецификацией:

$$\text{unit}(A : \text{Set}) := \forall f, g \in \text{Hom}(A, 1) (f =_{\text{Hom}(A, 1)} g).$$

Найти в выбранной системе представление терминального объекта, удовлетворяющее указанной спецификации.

Список литературы

1. *Claessen K., Hughes J.* QuickCheck: a lightweight tool for random testing of Haskell programs // Proceedings of the fifth ACM SIGPLAN international conference on Functional programming. — New York, NY, USA : Association for Computing Machinery, 09.2000. — С. 268—279. — (ICFP '00). — ISBN 978-1-58113-202-1. — DOI: 10.1145/351240.351266.
2. *Вольфенгаген В. Э.* Комбинаторная логика в программировании. Вычисления с объектами в примерах и задачах. — 3-е изд., дополн. и перераб. — М. : Институт «ЮрИнфоР МГУ», 2008.
3. *Маклейн С.* Категории для работающего математика. — Физматлит, 2004. — пер. с англ.