

Ю. Ю. Белых
(ГрГУ им. Я. Купалы, Гродно)
**ТЕОРИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ
И МЕТОДЫ ТРАНСЛЯЦИИ**

Под трансляцией в самом широком смысле можно понимать процесс восприятия компьютером программы, написанной на некотором формальном языке. При всем своем различии формальные языки имеют много общего и, в принципе, эквиваленты с точки зрения потенциальной возможности написать одну и ту же программу на любом из них. На самом деле сложно подвести под одну схему имеющееся многообразие языков программирования, Компиляция – преобразование объектов (данных и операций над ними) с входного языка в объекты на другом языке для всей программы в целом с последующим выполнением полученной программы в виде отдельного шага.

Интерпретация – анализ отдельного объекта на входном языке с одновременным выполнением (интерпретацией) [1].

Следовательно, компиляция и интерпретация отличаются не характером и методами анализа и преобразования объектов программы, а совмещением фаз обработки этих объектов во времени. То есть при компиляции фазы преобразования и выполнения действий разнесены во времени, но зато каждая из них выполняется над всеми объектами программы одновременно. При интерпретации, наоборот, преобразование и выполнение действий объединены во времени, но для каждого объекта программы.

Если посмотреть на эти различия несколько с другой стороны, то можно заметить, что интерпретатор непосредственно выполняет действия, связанные с определением или преобразованием объектов программы, а компилятор – переводит их на другой (не обязательно машинный язык).

Создание слоя программной интерпретации для некоторого промежуточного языка в практике построения трансляторов обычно встречается при попытке обеспечить совместимость для имеющегося многообразия языков программирования, операционных систем, архитектур и т.д. То есть определяется некоторый внутренний промежуточный язык, достаточно простой, чтобы для него можно было написать интерпретатор для всего имеющегося многообразия операционных систем или архитектур. Затем пишется один (или несколько) компиляторов для одного (или нескольких) входных языков на этот промежуточный уровень.

В рамках данной работы для изучения принципов теории языков программирования и методик построения компиляторов и трансляторов была поставлена задача разработать язык высокого уровня, включающий арифметические выражения (операции сложения, вычитания, умножения, деления, унарный плюс-минус, скобки); логические выражения (логические И, ИЛИ, НЕ, скобки); основные конструкции языка – присваивание; выбор; циклы; переключение; переход; составные операторы; функции и/или процедуры и оператор типа return; простые средства ввода-вывода. Данные не менее 2-х «встроенных» типов (например, целые и вещественные числа, литерные, логические и др.) и одного из пользовательских типов (объекты, структуры, объединения и др.) с операторами объявления.

Для достижения поставленной цели было принято решение реализовывать интерпретатор языка из семейства диалектов LISP. Лисп является вторым в истории (после Фортрана) используемым по сей день высокоуровневым языком программирования и имеет относительно малый набор синтаксических правил, которые, тем не менее, путем композиций дают языку огромную выразительную силу. Таким образом реализовав лишь необходимый минимум грамматических правил, основную функциональность языка можно реализовать на нем же самом.

Лисп – интерпретируемый язык, поэтому как целевая платформа была выбрана JavaScript runtime, таким образом интерпретатор должен быть написан на ECMAScript. Такой выбор позволит на практике показать идею слоя программной интерпретации для обеспечения кроссплатформенности языка. На данный момент существует большое количество различных реализаций сред выполнения JavaScript (Google V8, Rhino, Mozilla SpiderMonkey и т.д.). Таким образом разработанный язык способен работать на любой из этих платформ.

Реализация ядра интерпретатора заняла около 1500 строк JavaScript кода, весь код выложен в свободный доступ через репозиторий GitHub.

Литература

1. Ахо, А. Compilers: Principles, Techniques, and Tools. Part 2 / Альфред В. Ахо, Моника С. Лам, Рави Сети, Джеффри Д. Ульман – М.: Вильямс, 2008. – 640 С.