УДК 519.685.3

Мелентьев К.И. (3 курс, каф. СП, СПбГУ)

Григорьев С.В. (магистр ИТ, ст. преп. кафедры системного программирования СПбГУ)

РЕАЛИЗАЦИЯ БИБЛИОТЕКИ ПАРСЕР-КОМБИНАТОРОВ НА ОСНОВЕ АЛГОРИТМА GLL ДЛЯ ПЛАТФОРМЫ .NET

Традиционно, для создания синтаксических анализаторов используются генераторы синтаксических анализаторов, при этом используется особый язык спецификации грамматики, а семантические действия, как правило, описываются на другом языке программирования. Парсер-комбинаторы позволяют реализовать синтаксический анализатор и семантические действия на одном языке, на котором пишется дальнейшая обработка данных. Более того, этот подход обладает и другими преимуществами: естественность и понятность описания анализаторов, а также простота отладки, обеспечиваемая свойством модульности. Однако многие библиотеки парсер-комбинаторов (например, Parsec, FParsec), наследуя свойство нисходящих синтаксических анализаторов, не поддерживают леворекурсивные правила грамматик.

Целью работы является реализация библиотеки парсер-комбинаторов для платформы .NET (на языке F#), позволяющей строить синтаксические анализаторы по описанию произвольной контекстно-свободной грамматики.

Основная особенность реализованной библиотеки FsGll[6] заключается в снятии ограничений на использование леворекурсивных и неоднозначных грамматик за счет использования алгоритма Generalized LL[1]. Кроме того, реализованная библиотека позволяет работать со входным потоком абстрактного типа, то есть как с потоком символов, так и, например, с потоком лексем (в отличие от библиотек FPrasec[4] и gll-combinators[2,3], которые работают только с потоком символов).

Библиотека FsGll также реализует интерфейсы для создания инкрементальных синтаксических анализаторов, применимых, для анализа исходного кода в IDE или для анализа данных, поступающих порциями (например, по сети). Алгоритм был реализован двумя способами: с использованием изменяемых структур данных и с использованием неизменяемых. Первый вариант требует дополнительную память для частичного копирования структур, второй вариант не копирует лишние данные, но его производительность значительно ниже. Библиотека предоставляет интерфейсы для обоих вариантов алгоритма, благодаря чему пользователь может выбрать тот, что соответствует его потребностям.

Таким образом, в ходе работы была реализована библиотека парсер-комбинаторов на языке F#, а также проведено сравнение по объему предоставляемой функциональности и производительности с существующими библиотеками на платформах .NET (FParsec, XParsec[5]) и jvm (gll-combinators — прототип нашей библиотеки).

ЛИТЕРАТУРА:

[1] Scott E., Johnstone A. GLL Parsing //Electronic Notes in Theoretical Computer Science (ENTCS). – 2010. – Т. 253. – №. 7. – С. 177-189.

[2] Страница проекта gll-combinators на GitHub — URL: <https://github.com/djspiewak/gll-combinators>

[3] Spiewak D. Generalized Parser Combinators. – 2010. — URL: <http://www.cs.uwm.edu/~dspiewak/papers/generalized-parser-combinators.pdf>

[4] Страница проекта FParsec — URL: <http://www.quanttec.com/fparsec/>

[5] Страница проекта XParsec — URL: <https://github.com/corsis/XParsec>

[6] Страница проекта FsGll на GitHub — URL: <https://github.com/YaccConstructor/FsGll>