

Формальные языки

Анвар Тлямов

Предложить конкретный синтаксис для языка **М**. При помощи любого инструмента реализовать парсер для этого языка.

- Конкретный синтаксис должен быть описан человекочитаемо: грамматика для antlr, конечно, специфицирует конкретный синтаксис, но лучше все-таки добавить подробностей и объяснений, что именно вы хотели ею выразить.
- Парсер должен быть встроен в консольное приложение, принимающее на вход путь к входному файлу.
- В результате работы парсера на некорректном входе пользователь должен получить цивилизованное сообщение об ошибке (пробросить исключение наружу — плохая идея).
- В результате работы парсера на корректном входе должно получаться абстрактное синтаксическое дерево, которое соответствует описанию абстрактного синтаксиса ниже. Его можно либо вывести в консоль, либо сохранить в файл — как удобнее.
- Конкретный синтаксис может быть вообще любым, учитывайте, что парсить этот язык вам же.
- Необходимо написать тесты.

Абстрактный синтаксис языка **М**

Программа на языке **М** состоит из возможно пустого множества определений отношений, за которыми следует цель.

Определение отношения состоит из названия отношения, имен аргументов и тела.

Тело отношения — цель.

Цель — унификация двух термов; конъюнкция или дизъюнкция двух целей; связывание свободных переменных или вызов отношений.

X — имена переменных, C — имена конструкторов, R — имена отношений.

\mathcal{T}	$=$	$X \cup \{C(t_1, \dots, t_i) \mid t_j \in \mathcal{T}\}$	термы над переменными
\mathcal{G}	$=$	$\mathcal{T} \equiv \mathcal{T}$	унификация термов
		$\mathcal{G} \wedge \mathcal{G}$	конъюнкция целей
		$\mathcal{G} \vee \mathcal{G}$	дизъюнкция целей
		$\text{fresh } \{X\} . \mathcal{G}$	свободные переменные
		$R(t_1, \dots, t_i), t_j \in \mathcal{T}$	вызов отношения
\mathcal{P}	$=$	$\{R = \lambda x_1 \dots x_i . g; \} g$	программа

Рис. 1: Абстрактный синтаксис языка **М**