

Санкт-Петербургский государственный университет

Кафедра Системного программирования

Свитков Сергей Андреевич

Реализация поиска путей с  
КС-ограничениями в рамках библиотеки  
YC.QuickGraph

Курсовая работа

Научный руководитель:  
ст. преп, к. ф-м. н. Григорьев С. В.

Санкт-Петербург  
2016

# Оглавление

Аннотация	3
Введение	4
1. Обзор предметной области	6
2. Постановка задачи	7
3. Основная часть	8
4. Эксперименты	9
Заключение	10
Список литературы	11

## Аннотация

Большинство промышленных языков для написания запросов к графовым базам данных являются регулярными. Но регулярные языки не применимы в ряде задач, поэтому актуальным является создание контекстно-свободного (далее — КС) языка запросов. Существуют работы по этой теме, но они в основном теоретические. В данной работе рассматривается практическая реализация механизма контекстно-свободных запросов к ориентированным графам с помеченными ребрами для платформы .NET. Результатом работы является библиотека, предоставляющая набор функций для написания КС-запросов к графам. Полученные результаты могут быть применены в проектах, использующих C# или F#.

# Введение

Модель представления данных в виде ориентированных графов с метками на ребрах, имеет широкую область применения и используется в биоинформатике, социальных исследованиях (например, при представлении социальных графов), semantic web, при реализации графовых баз данных.

При наличии представления данных в виде определенной структуры становится актуальным вопрос их обработки, а именно — получения из всего набора только тех данных, которые представляют какой-либо конкретный интерес. Для этого используются языки запросов. Существует множество промышленных языков запросов к графам, например Gremlin[5], Cypher[2], и т.д.. Но данные языки являются регулярными, а значит, не могут применяться в некоторых задачах. Например, при разборе генеалогического дерева, встречаются строки вида  $parent^n child^n$ . Такие строки нельзя распознать с помощью регулярной грамматики, но можно с помощью КС-грамматики с правилами вывода  $N \rightarrow parent\ child, N \rightarrow parentN\ child$ .

Существуют работы, предлагающие различные подходы к реализации КС-запросов к графам, например [4], [1]. Но большая часть работ по данной теме представляет только теоретические сведения о возможных подходах к реализации, а те, что реализованы на практике, имеют довольно ограниченный функционал или же слишком узкую специализацию. Так, в работе [1] результатом запроса является КС-отношение — тройка вида  $(n, m, N)$ , где  $n$  и  $m$  — вершины, связанные путем, выводимым из нетерминала  $N$ .

Поскольку класс задач, в которых могут быть применены КС-запросы к графам, является обширным, то возникает задача создания инструмента, с помощью которого можно было бы получать результат КС-запроса в нескольких формах. Для этого требуется язык запросов и средства синтаксического анализа. Одним из алгоритмов синтаксического анализа является GLL[3]. Как описано в работе [9], данный алгоритм имеет широкое применение в задачах синтаксического анализа

и хорошую асимптотику ( $O(n^3)$  — на нелинейном входе,  $O(n)$  — на линейном). Кроме того, результатом вышеупомянутой статьи является реализация алгоритма в рамках проекта YaccConstructor [7] (далее — YC). Кроме того, YC имеет язык спецификаций грамматик YARD [8], который можно использовать в качестве языка запросов. Поэтому было принято решение использовать YC как инструмент для решения поставленной задачи. Средства для работы с графами собраны в библиотеке YC.QuickGraph [6].

Поэтому механизм поиска путей с КС-ограничениями был реализован как расширение библиотеки YC.QuickGraph с использованием языка YARD для задания грамматик. Полученный результат — библиотека, которая позволяет осуществлять КС-запросы к графам с помеченными ребрами, представляя результат в виде подграфа, пути или КС-отношения.

# 1. Обзор предметной области

1

## 2. Постановка задачи

1

### **3. Основная часть**

1



## 4. Эксперименты

1

# Заклучение

1

## Список литературы

- [1] Hellings Jelle. Conjunctive context-free path queries. — 2014.
- [2] Neo4j. Cypher // Neo4j official page. — URL: <https://neo4j.com/developer/cypher/> (online; accessed: 22.11.2016).
- [3] Scott Elizabeth, Johnstone Adrian. GLL parsing // Electronic Notes in Theoretical Computer Science. — 2010. — Vol. 253, no. 7. — P. 177–189.
- [4] Sevon Petteri, Eronen Lauri. Subgraph queries by context-free grammars // Journal of Integrative Bioinformatics. — 2008. — Vol. 5, no. 2. — P. 100.
- [5] Titan. Gremlin // Titan official page. — URL: <https://github.com/tinkerpop/gremlin/wiki> (online; accessed: 29.11.2016', language =).
- [6] YaccConstructor. YC.QuickGraph // YaccConstructor official page. — URL: <http://yaccconstructor.github.io/QuickGraph/> (online; accessed: 22.11.2016).
- [7] YaccConstructor. YaccConstructor // YaccConstructor official page. — URL: <http://yaccconstructor.github.io> (online; accessed: 29.11.2016).
- [8] YaccConstructor. YARD // YaccConstructor official page. — 2015. — URL: <http://yaccconstructor.github.io/YaccConstructor/yard.html> (online; accessed: 22.11.2016).
- [9] Рагозина Анастасия Константиновна, Шкредов СД. Ослабленный синтаксический анализ динамически формируемых программ на основе алгоритма GLL. — 2016.