Реализация алгоритма Кока-Янгера-Касами (CYK)

АВТОР: СУСАНИНА ЮЛИЯ АЛЕКСЕЕВНА, 143 ГРУППА
РУКОВОДИТЕЛЬ: ст.пр. ГРИГОРЬЕВ С. В.

МАТЕМАТИКО-МЕХАНИЧЕСКИЙ ФАКУЛЬТЕТ
САНКТ-ПЕТЕРБУРГСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА
26.09.2016

Введение

Алгоритм Кока-Янгера-Касами, или алгоритм CYK – это алгоритм, позволяющий определить возможен ли вывод строки в заданной грамматике в нормальной форме Хомского.

Другими словами, это алгоритм синтаксического анализа строки.

Алгоритм реализует синтаксический анализ снизу-вверх и основывается на методе динамического программирования.

Цели и задачи

Цель работы: синтаксический анализ строк (с использованием алгоритма CYK)

Задачи:

- Изучение алгоритма CYK и принципа его работы
- Реализация алгоритма на F#
- Параллельная реализация алгоритма на F#
- Реализация алгоритма на GPGPU
- Реализация тестов производительности
- Применение алгоритма для поиска подстрок в строке
- Разработка тестов для проверки корректности работы данного алгоритма

Алгоритм Кока-Янгера-Касами

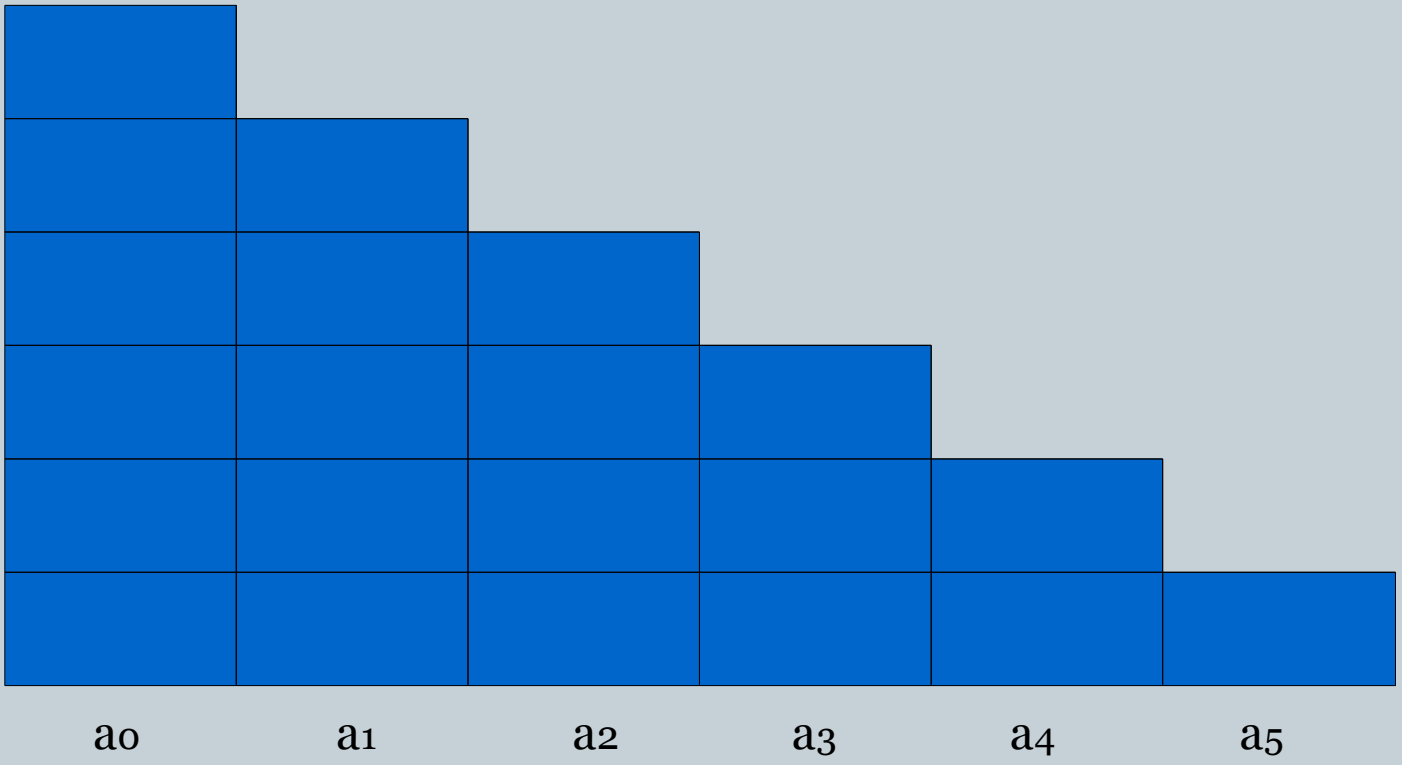
Данный алгоритм работает с контекстно-свободными грамматиками, имеющими нормальную форму Хомского.

Нормальная форма Хомского требует от грамматики, чтобы каждая ее продукция выглядела одним из трех способов :

1. $S \rightarrow AB$, где S – стартовый нетерминальный символ, A,B – нетерминальные символы,
2. $A \rightarrow a$ a – терминал, ε – пустая строка
3. $S \rightarrow \varepsilon$

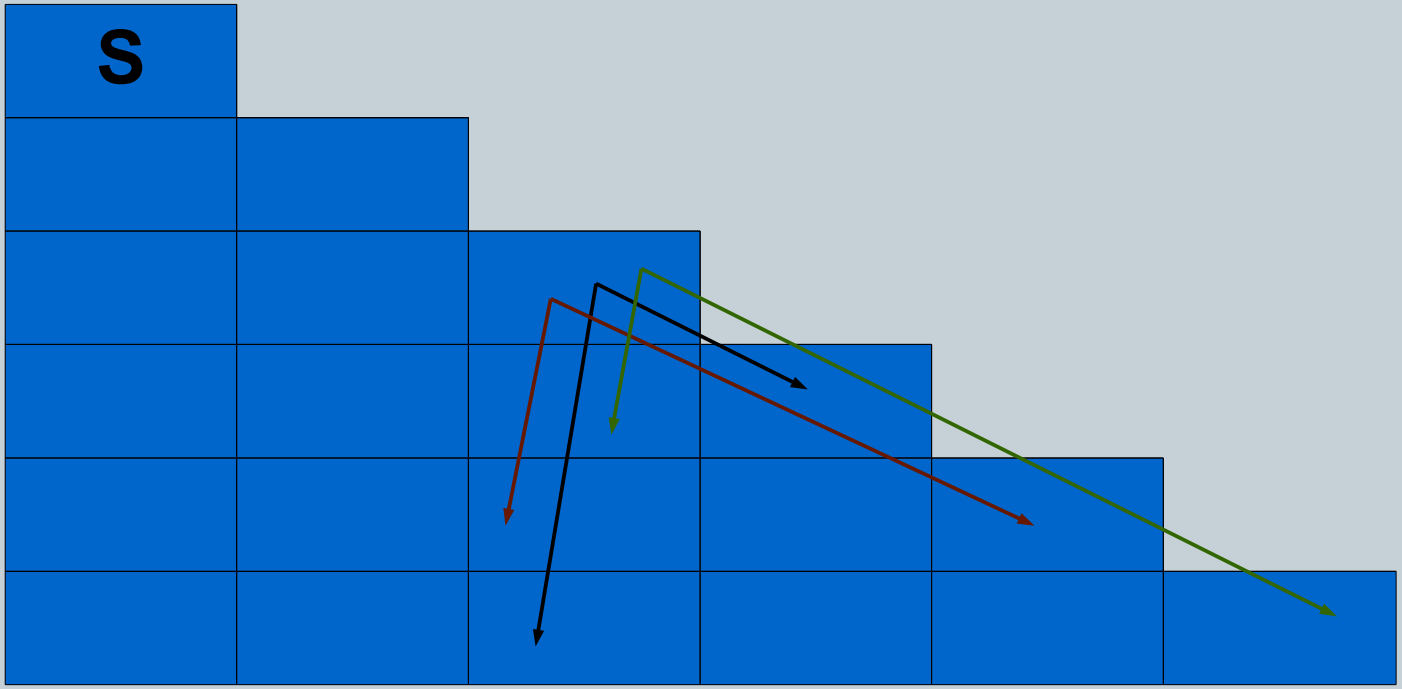
Алгоритм Кока-Янгера-Касами

Для применения алгоритма к входной строке ($a_1,\ldots,a_n$) строится треугольная матрица размера n*n, где n – длина входной строки.



Алгоритм Кока-Янгера-Касами

Если после заполнения всех элементов матрицы, если в вернем левом углу стоит стартовый нетерминал, то строка выводится из данной грамматики.

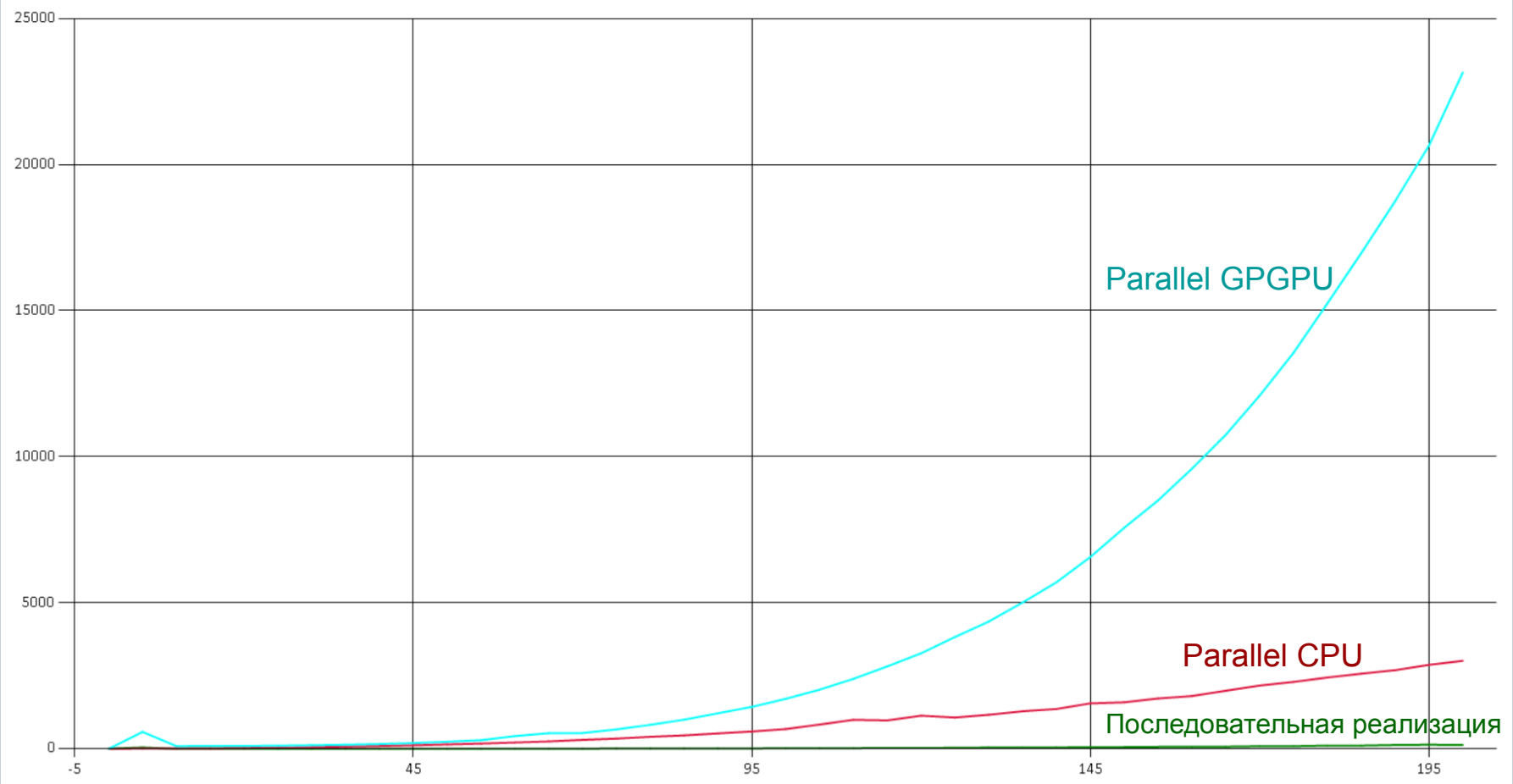


Реализация алгоритма

На языке программирования F#:

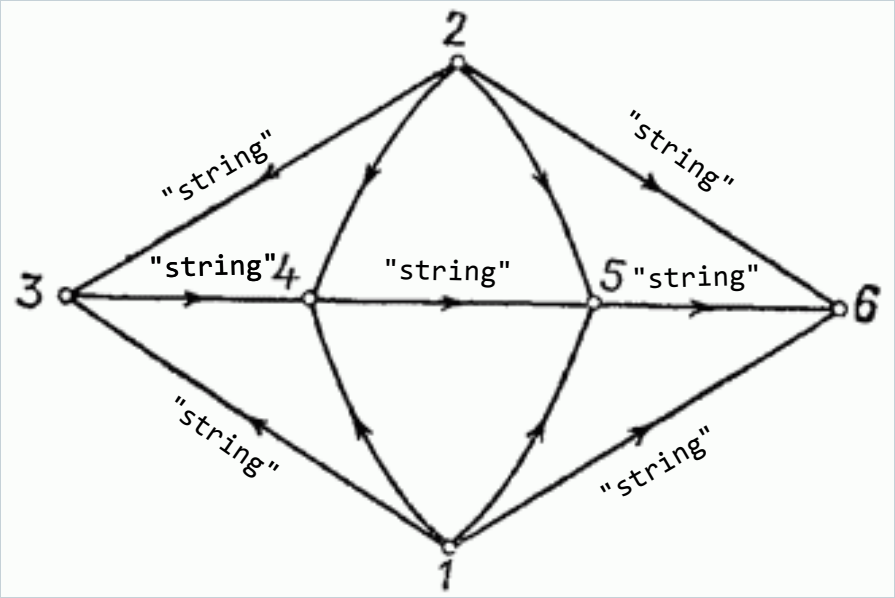
- Простая последовательная реализация
- Параллельная реализация (с использованием класса Parallel)
- Параллельная реализация на GPGPU (с использованием библиотеки Brahma.FSharp)

Реализация тестов производительности



Поиск подстрок в строке

Данный алгоритм был применен для поиска подстрок, указанной длины, выводимых из заданной грамматики.



Тестирование

Разработка тестов для проверки корректности работы алгоритма:

- Для строк, выводимых из заданной грамматики
- Для строк, невыводимых из заданной грамматики
- Случаев, когда пользователь вводит пустую строку

Разработка тестов для проверки правильности результата для поиска подстрок в строке

Результаты

- Изучен алгоритм CYK и принцип его работы
- Реализован алгоритм на F#
- Реализована параллельная версия алгоритма на F#
- Реализован алгоритм на GPGPU
- Реализованы тесты производительности
- Применен алгоритм для поиска подстрок в строке
- Разработаны тесты для проверки корректности работы данного алгоритма