

Efficiency of Generalised LL Parsing

Mini-Workshop 2IS95: SET Seminar

Jules Wolms

Computer Science and Mathematics
Eindhoven University of Technology

November 25 2014

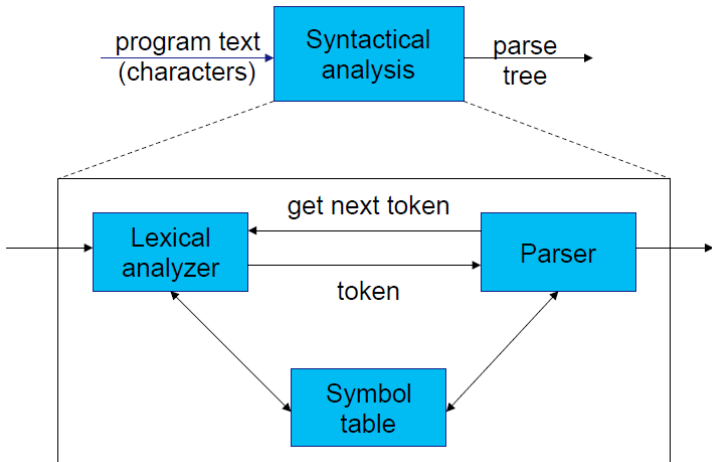
Outline

- 1 Introduction
- 2 Proposal
- 3 Discussion

Introduction to Topic

- Parsing
- (G)LL Parsing
- Motivation

Parsing



LL Parsing

Grammar

```
S ::= c A d  
A ::= a | b
```

Parser

```
bool proc S()  
  if "c"  
    then if A()  
          then if "d"  
                then return true  
                else return false  
  return false
```

```
bool proc A()  
  if "a"  
    then return true  
  else if "b"  
    then return true  
  else return false
```

GLL Parsing

Generalised LL parser

- Recursive descent parser
- Explicit control of stack → datastructure
- Multiple derivations in parallel → algorithm + datastructure

Motivation

- Interests in:
 - Software Engineering & Technology
 - Algorithms
- Parsing is very algorithmic
- Parsing is explored, but GLL parsing is new

Literature & Approach

Literature

Adrian Johnstone & Elizabeth Scott

Approach

- Find better datastructures/algorithms
- Analyse running time and memory usage
- Iterate & Combine

Questions

Any questions about the topic, literature or approach?