LR0 Table

Parsing

- It's a little awkward to use the DFA in the parse stage
- So we will build a parse table
  - Sort of like LL table
  - One row per state
  - One column per grammar symbol
- If you are in state s and next input symbol is t: Consult table[s][t] to determine what to do next
  - Recall: LR parser can either shift or reduce.

Build Table

- To build the table: Suppose we have a list of DFA states
- First, we assign each state a unique number:

```
for i in range(len(states)):
    states[i].index = i
```

  - Or, you could have assigned these numbers when you created the DFA nodes

- Now we're ready to build the table:
  table = [ ]

Build Table

```
for s in states:
    row = {}
    for sym in s.transitions:
        if sym is a terminal:
            row[sym] = ("S",s.transitions[sym].index)
        else:
            row[sym] = ("T",s.transitions[sym].index)
    for item in s.items:
        lhs, rhs, dpos = item
        if dpos at end of rhs:
            for w in terminals:
                row[w] = ("R", len(rhs), lhs )
            row["$"] = ("R", len(rhs), lhs )
    table.append(row)l
```

Conflict

- If table cell set more than once: Grammar is not LR(0)
  - Shift-reduce or reduce-reduce conflict

Shift-Reduce

- SR conflict: Parser could shift or reduce
  - Remember: Shift means “consume token from input” while reduce means “convert a RHS to a LHS”
- Often, we can resolve SR conflicts by preferring shift
  - But sometimes that’s not the right choice
  - Later, we’ll see how we can restrict parser’s choices to deal with conflicts

Reduce-Reduce

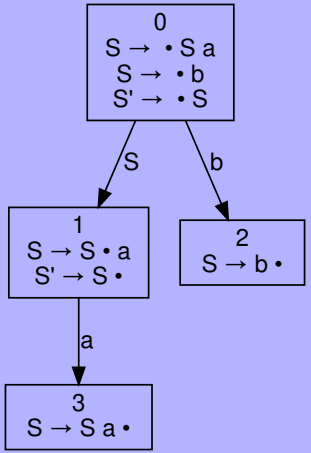
- Parser has two or more choices of items it can reduce to
- Ex: Suppose state has items:
  $A \rightarrow B\ C\ \bullet$
  $X \rightarrow Y\ \bullet$
- Should parser reduce to A or reduce to X?

Example

- Consider this grammar:

$$a \to a$$
$$b \to b$$
$$S \to S\ a \mid b$$

DFA



Table

	State	$	S	a	b
0	$S \rightarrow \bullet S\ a$ $S \rightarrow \bullet b$ $S' \rightarrow \bullet S$		T,1		S,2
1	$S \rightarrow S \bullet a$ $S' \rightarrow S \bullet$	R,1,S'		S,3 R,1,S'	R,1,S'
2	$S \rightarrow b \bullet$	R,1,S		R,1,S	R,1,S
3	$S \rightarrow S\ a \bullet$	R,2,S		R,2,S	R,2,S

Assignment

- None!
- Next time, we'll build and use the table...

Sources

- K. Louden. Compiler Construction: Principles and Practice .
- A. Aho, M. Lam, R. Sethi, J. Ullman. Compilers: Principles, Techniques, & Tools. (2nd ed.) Addison-Wesley.

Created using LaTeX.

Main font: Gentium Book Basic, by Victor Gaultney. See http://software.sil.org/gentium/
Monospace font: Source Code Pro, by Paul D. Hunt. See https://fonts.google.com/specimen/Source+Code+Pro and http://sourceforge.net/adobe
Icons by Ulisse Perusin, Steven Garrity, Lapo Calamandrei, Ryan Collier, Rodney Dawes, Andreas Nilsson, Tuomas Kuosmanen, Garrett LeSage, and Jakub Steiner. See http://tango-project.org