Definitions

- LR(1) stands for *Left-to-right* scan of input, *Right-canonical* derivation, *1* symbol of lookahead.
- An LR(1) parser is a *bottom-up* parser; it begins with the input string and finds a *reversed* derivation of the input, ending at the start symbol.
- An LR(1) grammar is a grammar that can be parsed by an LR(1) parser.

LR(1) Algorithm

```plaintext
push q0
for each a in ⊢ input ⊣
    while (Reduce[stack.top, a] = A → γ)
        pop 2*|γ| times
        state ← stack.top
        push A
        push Trans[state, A]
        state ← stack.top
        if (Trans[state, a] = ERROR) reject
        push a
    push Trans[state, a]
accept
```

Bottom-up Parsing (LR(1))

Consider the following context-free grammar and LR(1) shift/reduce table:

0. $S' \rightarrow ⊥ S ⊥$
1. $S \rightarrow Sab$
2. $S \rightarrow XY$
3. $X \rightarrow pX$
4. $X \rightarrow ε$
5. $Y \rightarrow q$
6. $Y \rightarrow ε$

- Using the shift/reduce table, parse the string ⊥pqab⊥. Write the reversed right-canonical derivation for the string, as well as the parse tree.
<table>
<thead>
<tr>
<th></th>
<th>⊢</th>
<th>shift 1</th>
<th>2</th>
<th>a</th>
<th>reduce 2</th>
<th>5</th>
<th>a</th>
<th>reduce 4</th>
<th>8</th>
<th>a</th>
<th>shift 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a</td>
<td>reduce 4</td>
<td>2</td>
<td>⊣</td>
<td>reduce 2</td>
<td>5</td>
<td>p</td>
<td>shift 5</td>
<td>8</td>
<td>⊣</td>
<td>shift 6</td>
</tr>
<tr>
<td>1</td>
<td>p</td>
<td>shift 5</td>
<td>3</td>
<td>a</td>
<td>reduce 6</td>
<td>5</td>
<td>q</td>
<td>reduce 4</td>
<td>9</td>
<td>a</td>
<td>reduce 3</td>
</tr>
<tr>
<td>1</td>
<td>q</td>
<td>reduce 4</td>
<td>3</td>
<td>q</td>
<td>shift 7</td>
<td>5</td>
<td>⊣</td>
<td>reduce 4</td>
<td>9</td>
<td>q</td>
<td>reduce 3</td>
</tr>
<tr>
<td>1</td>
<td>⊣</td>
<td>reduce 4</td>
<td>3</td>
<td>⊣</td>
<td>reduce 6</td>
<td>5</td>
<td>X</td>
<td>shift 9</td>
<td>9</td>
<td>⊣</td>
<td>reduce 3</td>
</tr>
<tr>
<td>1</td>
<td>S</td>
<td>shift 8</td>
<td>3</td>
<td>Y</td>
<td>shift 2</td>
<td>7</td>
<td>a</td>
<td>reduce 5</td>
<td>10</td>
<td>a</td>
<td>reduce 1</td>
</tr>
<tr>
<td>1</td>
<td>X</td>
<td>shift 3</td>
<td>4</td>
<td>b</td>
<td>shift 10</td>
<td>7</td>
<td>⊣</td>
<td>reduce 5</td>
<td>10</td>
<td>⊣</td>
<td>reduce 1</td>
</tr>
</tbody>
</table>