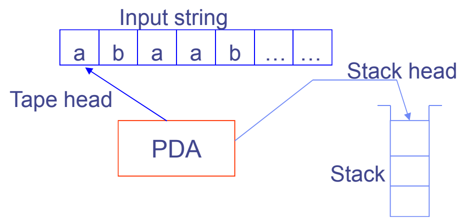
CSC4510 AUTOMATA

## 5.1 Definitions and Examples

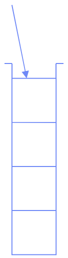
## 5.2 Deterministic Pushdown Automata

## Definitions and Examples

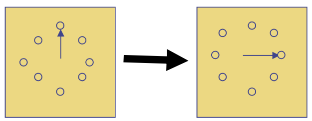
- A language can be generated by a CFG if and only if it can be accepted by a *pushdown automaton.*
- A pushdown automaton is similar to an FA but has an auxiliary memory in the form of a stack.
- Pushdown automata are, by default, nondeterministic. Unlike FA's, the nondeterminism cannot always be removed.



## Definitions and Examples (cont'd.)



- The stack head always scans the top symbol of the stack. It performs two basic operations:
  - Push: add a new symbol at the top.
  - Pop: read and remove the top symbol.
- Alphabet of stack symbols: Γ

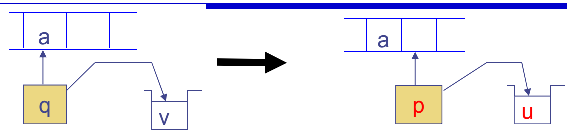


- A PDA has finitely many states which form a set *Q*. For each move, the state is changed according to the evaluation of a *transition function*

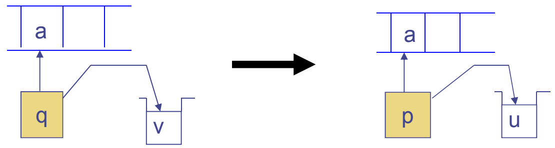
  $\delta: Q \times (\Sigma \cup \{\Lambda\}) \times \Gamma \rightarrow Q \times \Gamma^*$.



## Definitions and Examples (cont'd.)

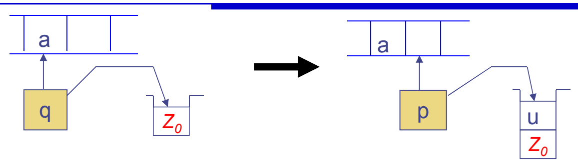



- $(p, u) \in \delta(q, a, v)$ means that if the tape head reads symbol *a*, the stack head reads symbol *v*, and the PDA is in state *q*, then one of the possible moves is that the next state is *p*, *v* is replaced by *u* at stack, and the tape head moves one cell to the right.

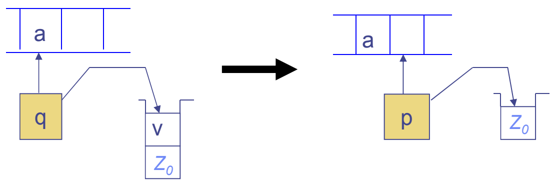



- $(p, u) \in \delta(q, \Lambda, v)$ means this is a Λ-transition.

## Definitions and Examples (cont'd.)



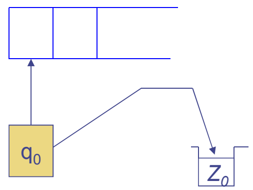

- *$(p, u) \in \delta(q, a, Z_0)$* means that a push operation is performed on the stack.




- *$(p, \Lambda) \in \delta(q, a, v)$* means that a pop operation is performed on the stack.



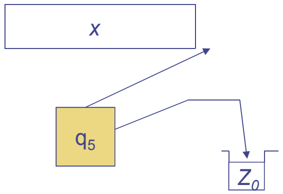
## Definitions and Examples (cont'd.)




- There are some special states: an initial state $q_0$ and a set $A$ of accepting states.

- Initially, the PDA is in the initial state $q_0$ and the head scans the leftmost cell. The tape holds an input string. The stack is empty with the initial stack symbol $Z_0$.

## Definitions and Examples (cont'd.)




- When the tape head gets off the tape, the PDA stops. An input string $x$ is accepted by the PDA if the PDA stops at an accepting state (and the stack is empty).
- Otherwise, the input string is rejected.



## Definitions and Examples (cont'd.)

- A single move of a PDA depends on
  1. current state
  2. next input
  3. symbol currently on top of the stack
- A PDA can replace the top symbol $X$ by a string $\alpha$ of stack symbols.
- Special cases are pushing a symbol $Y$ (replacing $X$ by $YX$) and popping $X$ (replacing $X$ by $\Lambda$)

## Definitions and Examples (cont'd.)

- Definition 5.1: A *pushdown automaton* (PDA) is a 7-tuple $M=(Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$, where:
  - $Q$ is a finite set of states
  - The *input* and *stack* alphabets $\Sigma$ and $\Gamma$ are finite sets
  - $q_0 \in Q$ is the initial state
  - $Z_0 \in \Gamma$ is the initial stack symbol
  - $A \subseteq Q$ is the set of accepting states
  - The transition function is $\delta$: $Q \times (\Sigma \cup \{\Lambda\}) \times \Gamma \rightarrow$ the set of finite subsets of $Q \times \Gamma^*$
- Because values of $\delta$ are sets, $M$ may be nondeterministic
- A move requires that there be at least one symbol on the stack. $Z_0$ is the one on the stack initially.



## Definitions and Examples (cont'd.)

- A *configuration* of a PDA is a triple $(q, x, \alpha)$
  - $q \in Q$ is the current state
  - $x \in \Sigma^*$ is the portion of the input string that has not yet been read
  - The contents of the stack is $\alpha \in \Gamma^*$
- $(p, x, \alpha) \vdash_M (q, y, \beta)$: one of the possible moves in the first configuration takes $M$ to the second.
  - $\vdash_M^n$ and $\vdash_M^*$ refer to $n$ moves and zero or more moves.

## Definitions and Examples (cont'd.)

- Definition 5.2: If $M=(Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ and $x \in \Sigma^*$, the string $x$ is *accepted* by $M$ if $(q_0, x, Z_0) \vdash_M^* (q, \Lambda, \alpha)$ for some $\alpha \in \Gamma^*$ and some $q \in A$
  - A language $L$ is said to be accepted by $M$ if $L$ is precisely the set of strings accepted by $M$.
  - Sometimes a string accepted by $M$ is said to be accepted *by final state*, because acceptance does not depend on the final stack contents.



## Definitions and Examples (cont'd.)

- $L(G)=AnBn = \{a^n b^n \mid n \geq 0\}$
- As soon as the PDA reads a '*b*':
  - It enters a new state in which only *b*'s are legal inputs.
  - It pops one *a* off the stack to cancel this *b*.
- The stack has no limit to its size, so the PDA can handle anything in *AnBn*.

## Definitions and Examples (cont'd.)

- A PDA for *AnBn* is $M=(Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ where $Q=\{q_0, q_1, q_2, q_3\}$, $A=\{q_0, q_3\}$, and the transitions are:

| Move # | State | Input | Stack Symbol | Move(s) |
|---|---|---|---|---|
| 1 | $q_0$ | $a$ | $Z_0$ | $(q_1, aZ_0)$ |
| 2 | $q_1$ | $a$ | $a$ | $(q_1, aa)$ |
| 3 | $q_1$ | $b$ | $a$ | $(q_2, \Lambda)$ |
| 4 | $q_2$ | $b$ | $a$ | $(q_2, \Lambda)$ |
| 5 | $q_2$ | $\Lambda$ | $Z_0$ | $(q_3, Z_0)$ |
| (all other combinations) | | | | none |

$$\begin{aligned}(q_0, aabb, Z_0) &\vdash (q_1, abb, aZ_0)\\ &\vdash (q_1, bb, aaZ_0)\\ &\vdash (q_2, b, aZ_0)\\ &\vdash (q_2, \Lambda, Z_0)\\ &\vdash (q_3, \Lambda, Z_0)\end{aligned}$$



## Definitions and Examples (cont'd.)

- A PDA for *SimplePal* is $M=(Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ where $Q=\{q_0, q_1, q_2\}$, $A=\{q_2\}$, and the transitions are:

| Move # | State | Input | Stack Symbol | Move(s) |
|---|---|---|---|---|
| 1 | $q_0$ | $a$ | $Z_0$ | $(q_0, aZ_0)$ |
| 2 | $q_0$ | $b$ | $Z_0$ | $(q_0, bZ_0)$ |
| 3 | $q_0$ | $a$ | $a$ | $(q_0, aa)$ |
| 4 | $q_0$ | $b$ | $a$ | $(q_0, ba)$ |
| 5 | $q_0$ | $a$ | $b$ | $(q_0, ab)$ |
| 6 | $q_0$ | $b$ | $b$ | $(q_0, bb)$ |
| 7 | $q_0$ | $c$ | $Z_0$ | $(q_1, Z_0)$ |
| 8 | $q_0$ | $c$ | $a$ | $(q_1, a)$ |
| 9 | $q_0$ | $c$ | $b$ | $(q_1, b)$ |
| 10 | $q_1$ | $a$ | $a$ | $(q_1, \Lambda)$ |
| 11 | $q_1$ | $b$ | $b$ | $(q_1, \Lambda)$ |
| 12 | $q_1$ | $\Lambda$ | $Z_0$ | $(q_2, Z_0)$ |
| (all other combinations) | | | | none |

$$(q_0, abcba, Z_0) \vdash (q_0, bcba, aZ_0) \vdash (q_0, cba, baZ_0) \vdash (q_1, ba, baZ_0)$$
$$\vdash (q_1, a, aZ_0) \vdash (q_1, \Lambda, Z_0) \vdash (q_2, \Lambda, Z_0)$$

## Definitions and Examples (cont'd.)

| Move # | State | Input | Stack Symbol | Move(s) |
|---|---|---|---|---|
| 1 | $q_0$ | $a$ | $Z_0$ | $(q_0, aZ_0)$ |
| 2 | $q_0$ | $b$ | $Z_0$ | $(q_0, bZ_0)$ |
| 3 | $q_0$ | $a$ | $a$ | $(q_0, aa)$ |
| 4 | $q_0$ | $b$ | $a$ | $(q_0, ba)$ |
| 5 | $q_0$ | $a$ | $b$ | $(q_0, ab)$ |
| 6 | $q_0$ | $b$ | $b$ | $(q_0, bb)$ |
| 7 | $q_0$ | $c$ | $Z_0$ | $(q_1, Z_0)$ |
| 8 | $q_0$ | $c$ | $a$ | $(q_1, a)$ |
| 9 | $q_0$ | $c$ | $b$ | $(q_1, b)$ |
| 10 | $q_1$ | $a$ | $a$ | $(q_1, \Lambda)$ |
| 11 | $q_1$ | $b$ | $b$ | $(q_1, \Lambda)$ |
| 12 | $q_1$ | $\Lambda$ | $Z_0$ | $(q_2, Z_0)$ |
| (all other combinations) | | | | none |

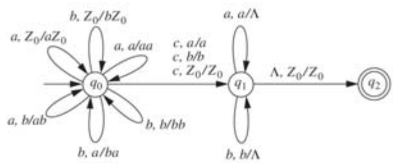
$$(q_0, acab, Z_0) \vdash (q_0, cab, aZ_0) \vdash (q_1, ab, aZ_0) \vdash (q_1, \Lambda b, Z_0) \vdash (q_2, b, Z_0)$$

$$(q_0, abc, Z_0) \vdash (q_0, bc, aZ_0) \vdash (q_0, c, baZ_0) \vdash (q_1, \Lambda, baZ_0)$$

15

## Definitions and Examples (cont'd.)

| Move # | State | Input | Stack Symbol | Move(s) |
|---|---|---|---|---|
| 1 | $q_0$ | $a$ | $Z_0$ | $(q_0, aZ_0)$ |
| 2 | $q_0$ | $b$ | $Z_0$ | $(q_0, bZ_0)$ |
| 3 | $q_0$ | $a$ | $a$ | $(q_0, aa)$ |
| 4 | $q_0$ | $b$ | $a$ | $(q_0, ba)$ |
| 5 | $q_0$ | $a$ | $b$ | $(q_0, ab)$ |
| 6 | $q_0$ | $b$ | $b$ | $(q_0, bb)$ |
| 7 | $q_0$ | $c$ | $Z_0$ | $(q_1, Z_0)$ |
| 8 | $q_0$ | $c$ | $a$ | $(q_1, a)$ |
| 9 | $q_0$ | $c$ | $b$ | $(q_1, b)$ |
| 10 | $q_1$ | $a$ | $a$ | $(q_1, \Lambda)$ |
| 11 | $q_1$ | $b$ | $b$ | $(q_1, \Lambda)$ |
| 12 | $q_1$ | $\Lambda$ | $Z_0$ | $(q_2, Z_0)$ |
| (all other combinations) | | | | none |




16

## Definitions and Examples (cont'd.)

- A PDA for *Pal* is $M=(Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ where $Q=\{q_0, q_1, q_2\}$, $A=\{q_2\}$, and the transitions are:

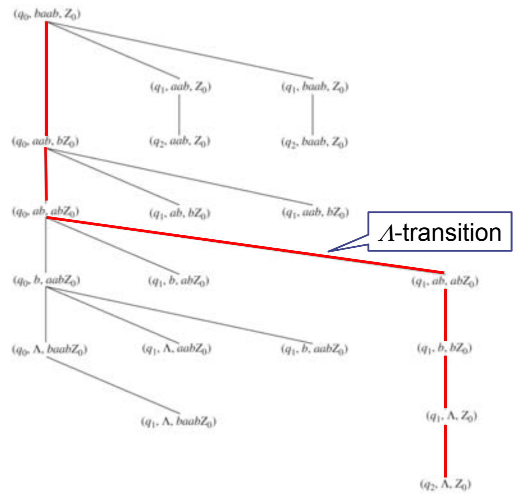
| Move # | State | Input | Stack Symbol | Move(s) |
|---|---|---|---|---|
| 1 | $q_0$ | $a$ | $Z_0$ | $(q_0, aZ_0), (q_1, Z_0)$ |
| 2 | $q_0$ | $a$ | $a$ | $(q_0, aa), (q_1, a)$ |
| 3 | $q_0$ | $a$ | $b$ | $(q_0, ab), (q_1, b)$ |
| 4 | $q_0$ | $b$ | $Z_0$ | $(q_0, bZ_0), (q_1, Z_0)$ |
| 5 | $q_0$ | $b$ | $a$ | $(q_0, ba), (q_1, a)$ |
| 6 | $q_0$ | $b$ | $b$ | $(q_0, bb), (q_1, b)$ |
| 7 | $q_0$ | $\Lambda$ | $Z_0$ | $(q_1, Z_0)$ |
| 8 | $q_0$ | $\Lambda$ | $A$ | $(q_1, a)$ |
| 9 | $q_0$ | $\Lambda$ | $b$ | $(q_1, b)$ |
| 10 | $q_1$ | $a$ | $a$ | $(q_1, \Lambda)$ |
| 11 | $q_1$ | $b$ | $b$ | $(q_1, \Lambda)$ |
| 12 | $q_1$ | $\Lambda$ | $Z_0$ | $(q_2, Z_0)$ |
| (all other combinations) | | | | none |

$$(q_0, abbba, Z_0) \vdash (q_0, bbba, aZ_0) \vdash (q_0, bba, baZ_0) \vdash (q_1, ba, baZ_0)$$
$$\vdash (q_1, a, aZ_0) \vdash (q_1, \Lambda, Z_0) \vdash (q_2, \Lambda, Z_0)$$



## Definitions and Examples (cont'd.)

- A computation tree for the input string *baab*



## Deterministic Pushdown Automata

- Definition 5.10: A pushdown automaton $M=(Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ is *deterministic* if it satisfies both of the following conditions:
  - For every $q \in Q$, every $\sigma$ in $\Sigma \cup \{\Lambda\}$, and every $X \in \Gamma$, the set $\delta(q, \sigma, X)$ has at most one element.
  - For every $q \in Q$, every $\sigma \in \Sigma$, and every $X \in \Gamma$, the two sets $\delta(q, \sigma, X)$ and $\delta(q, \Lambda, X)$ cannot both be nonempty.
- A language $L$ is a *deterministic context-free language* (DCFL) if there is a deterministic PDA (DPDA) accepting $L$.



## Deterministic Pushdown Automata (cont'd.)

- One example is the previous PDA accepting *AnBn*
- Another example: the language of balanced strings of brackets
  - Two states $q_0$ and $q_1$, where $q_0$ is the accepting state
  - Input symbols are [ and ]
  - Stack symbols are $Z_0$ and [

| Move # | State | Input | Stack Symbol | Move(s) |
|---|---|---|---|---|
| 1 | $q_0$ | [ | $Z_0$ | $(q_1, [Z_0)$ |
| 2 | $q_1$ | [ | [ | $(q_1, [[)$ |
| 3 | $q_1$ | ] | [ | $(q_1, \Lambda)$ |
| 4 | $q_1$ | $\Lambda$ | $Z_0$ | $(q_0, Z_0)$ |
| (all other combinations) | | | | none |

## Deterministic Pushdown Automata (cont'd.)

- The language *Pal* of palindromes over $\{a, b\}$ can be accepted by a PDA $M$ that saves symbols on the stack until it "guesses" that it has reached the middle of the string, then cancels stack symbols with input symbols.
- The initial state of $M$, in which it stays while it is processing the first half of the string, is $q_0$.
- The state it enters when it is ready to begin the second half is $q_1$, and the accepting state is $q_2$.
- Typical lines from the transition table:
  $(q_0, a, Z_0) = \{(q_0, aZ_0), (q_1, Z_0)\}$
  $(q_0, a, b) = \{(q_0, ab), (q_1, b)\}$
  $(q_0, \Lambda, b) = \{(q_1, b)\}$



## Deterministic Pushdown Automata (cont'd.)

- $(q_0, aba, Z_0) \vdash (q_0, ba, aZ_0) \vdash (q_1, a, aZ_0) \vdash (q_1, \Lambda, Z_0) \vdash (q_2, \Lambda, Z_0)$
  $(q_0, aa, Z_0) \vdash (q_0, a, aZ_0) \vdash (q_1, a, aZ_0) \vdash (q_1, \Lambda, Z_0) \vdash (q_2, \Lambda, Z_0)$
- The only nondeterminism is in the transition from $q_0$ to $q_1$.
- Theorem 5.16: The language *Pal* cannot be accepted by a DPDA (*i.e.*, cannot be accepted without guessing).