# Software Specifications
# Push Down Automatons

Cain Susko

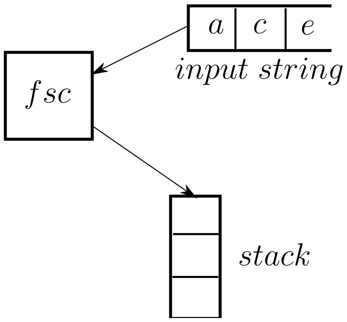
Queen's University
School of Computing

February 7, 2022

# Machine Model for Context Free Grammars

A grammar is whats known as a generative device. Because of this, compilers need *parsers* that are recognition devices for to understand context free languages.

This recognition device is whats known as a Push-down Automaton. Below is a visual representation of a Pushdown Automaton.

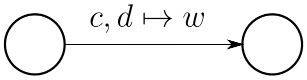


This finite state machine has a Finite State Control ($fsc$) which has access to a pushdown stack. The $fsc$ decides the next transition based on its current state as well as the scanned input symbol and the top of the stack.

The computation begins with an empty stack and the condition for accepting an ending state is that the stack must be empty at the end of the computation. However different types of Pushdown Automaton may have different accepting conditions.

The notation for a Pushdown Automaton (PDA) is as follows:

| | |
|---|---|
| $\Sigma$ | Input Alphabet |
| $\Gamma$ | Stack Alphabet |



Where $c \in \Sigma \cup \{\epsilon\}$ and $\mapsto$ signifies a stack operation, thus $d \in \Gamma \cup \{\epsilon\}$ and $w$ is any string over the stack alphabet: $w \in \Gamma*$
This transition between states signifies that

1. read $c$ from input

2. pop $d$ from stack

3. push $w$ to stack

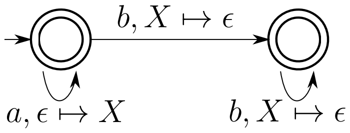
Note that $w$ can be a string and thus we can push multiple characters to the stack in one transition. Furthermore, it is recommended for the sake of readability that $\Sigma \cap \Gamma = \emptyset$. Which is to say that the input and stack alphabets do not share any characters.

# Example

Given the set of balanced strings:

$$L = \{a^i b^i \mid i \geq 0\}.$$

Determine a corresponding Pushdown Automaton.



Using a table we can visualize this pushdown automata in action with the input of $a^3b^3$:

| state | stack | remaining input |
|---|---|---|
| 1 | $\epsilon$ | aaabbb |
| 1 | X | aabbb |
| 1 | XX | abbb |
| 1 | XXX | bbb |
| 2 | XX | bb |
| 2 | X | b |
| 2 | $\epsilon$ | $\epsilon$ |

And thus the input is accepted as the automata is in a final state (all states are final in the example) and the stack is empty. Note that is convention for the top of the stack to be the leftmost part of the stack (like in the table)