Table of Contents

Introduction

A pushdown automaton (PDA) is a type of automaton that employs a stack (Pushdown automaton, n.d.). The term "pushdown" refers to the fact that the stack can be regarded as being "pushed down" like a tray dispenser at a cafeteria, since the operations are done only with the top element. It is an extension of the nondeterministic finite with e-transitions, which is one of the ways to define the regular languages. It is simply an *e*-NFA with the addition of a stack. The stack can be read, pushed, and popped only at the top and is similar with data structure's stack (Ullman & Hopcroft, 2001). Just as a deterministic finite automaton (DFA) is a way to implement a regular expression, a PDA is a way to implement a context free grammar (CFG) (Pushdown Automata, n.d.). Basically, its language-defining power is equivalent to that of a CFG (Pushdown Automata - The Stanford University InfoLab).

The presence of a stack in PDA means that, unlike the finite automaton, the pushdown automaton can "remember" an infinite amount of information. But unlike computers that have the ability to store or remember large amounts of information, the PDA can only access the information on its stack in a last-in-first-out manner. (Ullman & Hopcroft, 2001)

A PDA is formally defined as a 7-tuple:

$M = (Q,\ \Sigma,\ \Gamma,\ \delta,\ q_0,\ Z,\ F)$ where

- Q is a finite set of *states*
- Σ is a finite set which is called the *input alphabet*
- Γ is a finite set which is called the *stack alphabet*
- δ is a finite subset of $Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \times Q \times \Gamma^*$, the *transition relation*.
- $q_0 \in Q$ is the *start state*
- $Z \in \Gamma$ is the *initial stack symbol*
- $F \subseteq Q$ is the set of *accepting states*

(Gopalakrishnan, 2011)

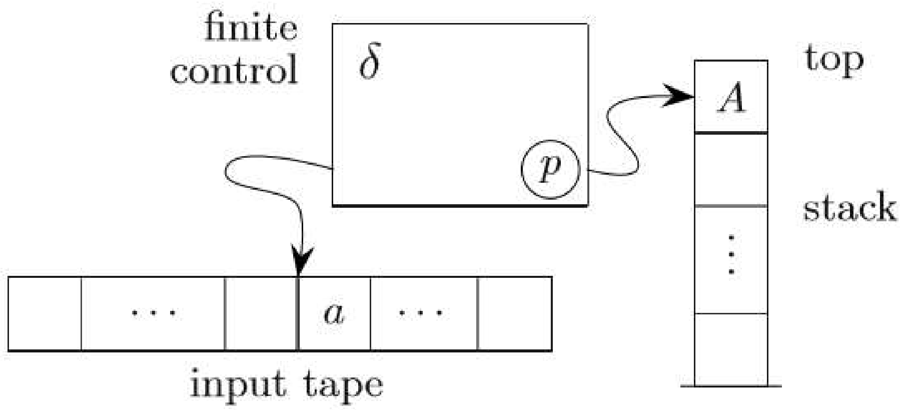


Figure 1.0 Pushdown automaton diagram (Hoogeboom & Engelfriet, 2004)

The transition path that is chosen is determined by the three parameters – input signal, current state ("p") and the symbol at the top of the stack, as shown in in figure 1.1 above. The automata can manipulate the stack by pushing a particular symbol to the top, or popping (removing) the topmost element. It can also ignore the stack, leaving it as it is. These manipulations are determined by the given transition table. (Pushdown Automata, 2003)

Project

For this project, a stack of bags displayed in a department store is simulated using the principles of pushdown automata based on a given CFG. A PDA has three operations namely push, pop, and replace (Mukherjee, 2013). The push operation is used whenever a new bag is added. Pop is used when a customer purchases a bag, which is then removed from the stack. Replace is implemented when the bag in front is placed back in the stock room, and replaced with another bag.

p = Pink Bag

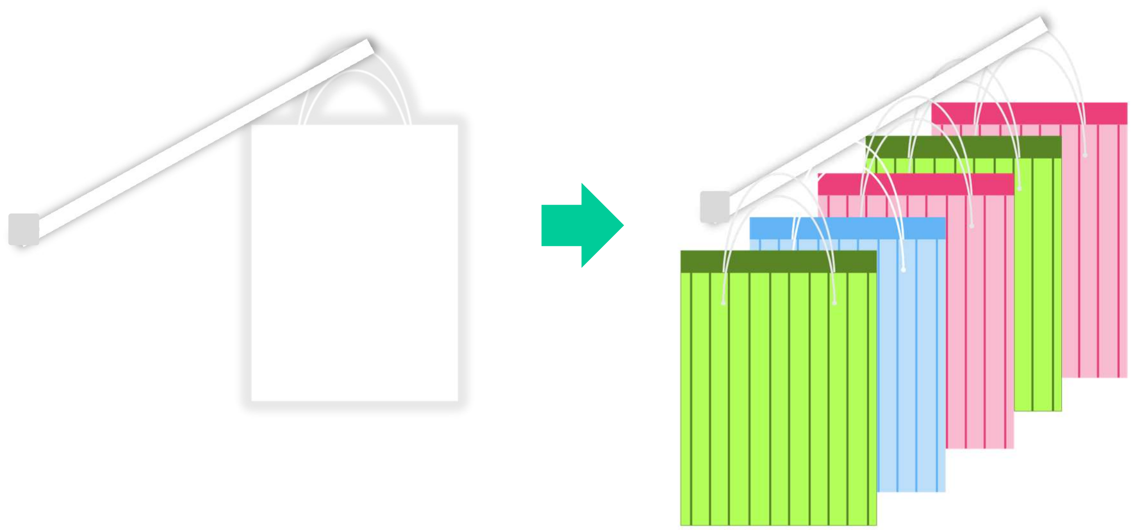
b = Blue Bag

g = Green Bag

$$S \rightarrow p \mid b \mid g \mid pS \mid bS \mid gS \mid \varepsilon$$

$$\delta(q_0, S, Z_0) = \{(q_1, S)\}$$

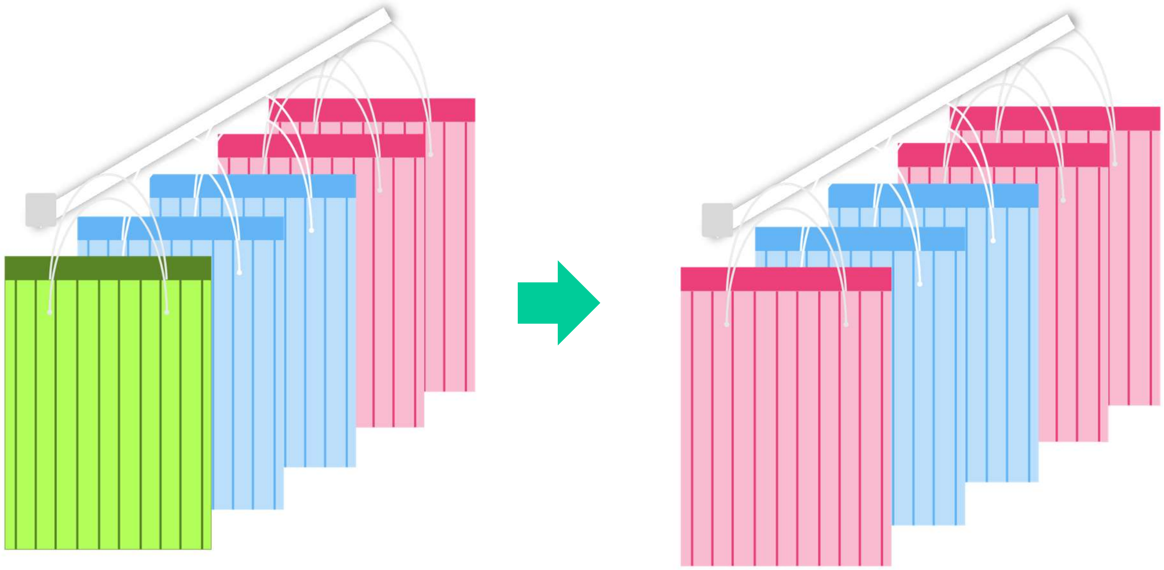
$$\delta(q_0, \varepsilon, S)$$
$$= \{(q_1, p), (q_1, b), (q_1, g), (q_1, pS), (q_1, bS), (q_1, gS), (q_1, \varepsilon)\}$$



In this scenario, the stack consists of blue, green and pink bags. Initially, the pink bags were placed in front, thus receiving more exposure to customers. They were selling out fast due to great demand, thereby causing the blue and green bags to be in front of the display, or in other words, on top of the stack. This caused the decrease in sales because of the blue and green bags are not as popular as the pink ones. In order to sell more products, the management should decide that whenever they get new stocks of pink bags, blue and green bags on top of the stack will be replaced with pink bags, by using the "replace" operation of PDA.

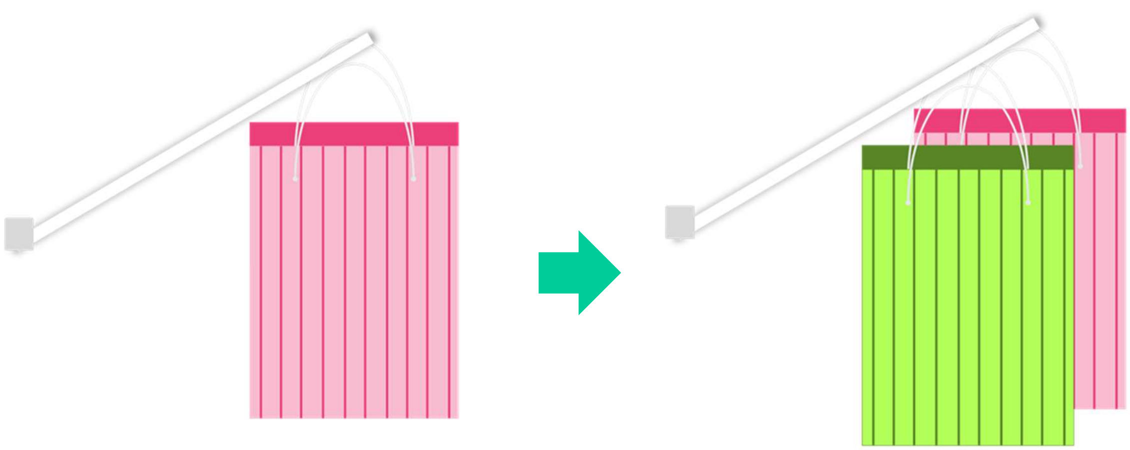
$$\delta(q_0, \varepsilon, g) = \{(q_1, p)\}$$

$$\delta(q_0, \varepsilon, b) = \{(q_1, p)\}$$

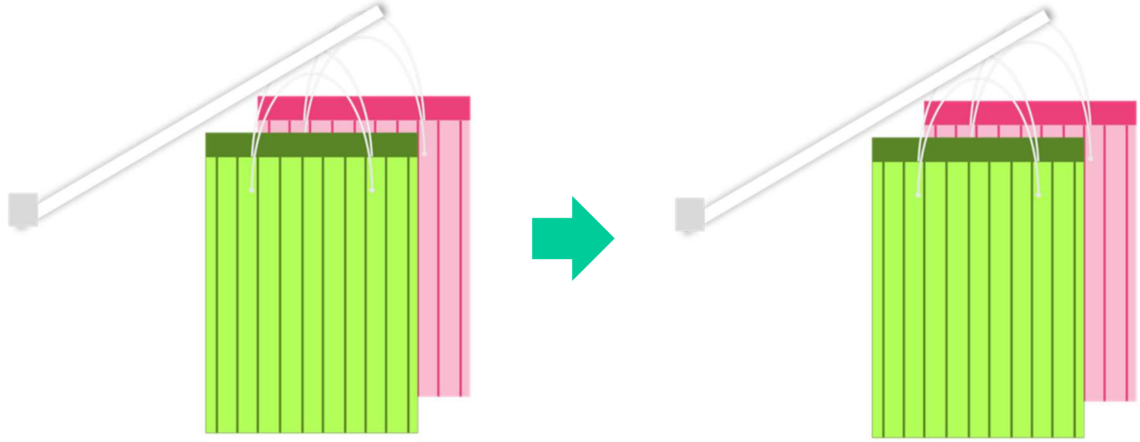


In this next scenario, the bags are becoming a huge hit so the management decides to add more color variations of the bags. This demonstrates the "push" operation. Since the grammar only accepts *p*, *b* and *g,* to put it in the project's context, the department store's line of bags must only consist of pink, blue and green bags. Other colors will be rejected and be considered as fake variants and will not be displayed.

$\delta(q_0, g, Z_1) = \{(q_1, gZ_1)\}$ (Z_1 = Current top of the stack)

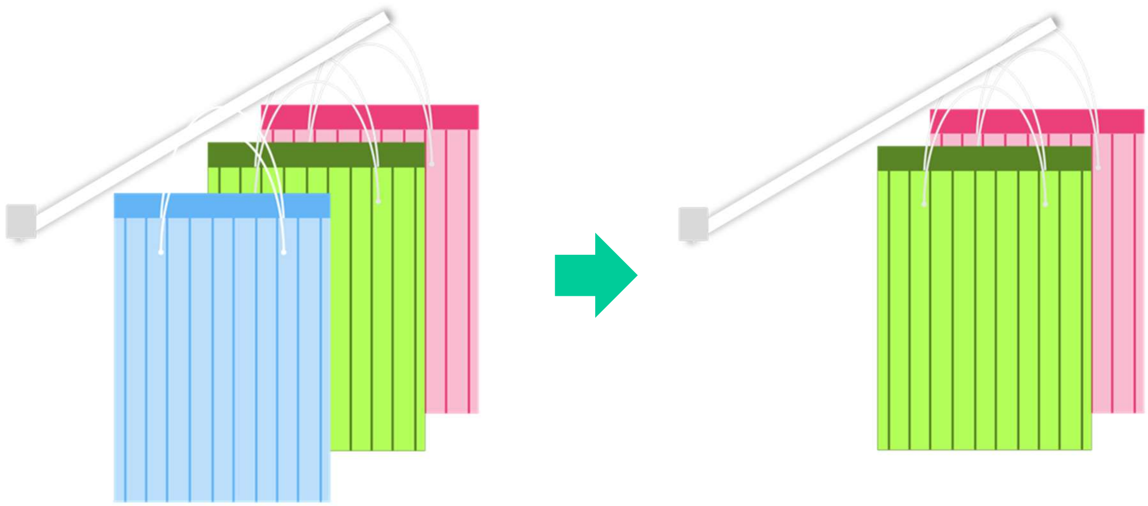


$\delta\left(q_0, v, Z_1\right) = \{\left(q_0, Z_1\right)\}$ (v = Violet bag)



Lastly, here's what happens when a customer purchases a bag, implementing the "pop" operation of PDA.

$\delta(q_0, b, b)=\{(q_1, \varepsilon)\}$



Bibliography

Ullman, J., & Hopcroft, J. (2001). *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley.

Gopalakrishnan, G. (2011). Notes on Push-down Automata. Utah. Retrieved September 27, 2015

Hoogeboom, H. J., & Engelfriet, J. (2004). Pushdown Automata. Tarragona, Spain. Retrieved September 27, 2015, from http://liacs.leidenuniv.nl/~hoogeboomhj/praatjes/tarragona/pda-intro.pdf

Mukherjee, D. (2013, February 8). Automata Theory : Push Down Automata Tutorial (PDA) Part 1. Retrieved from https://www.youtube.com/watch?v=QWX9CwiukL4

Pushdown Automata. (n.d.). Retrieved September 27, 2015, from www.cs.rit.edu/~jmg/courses/cs380/20031/slides/pda1-1.pdf

Pushdown Automata. (n.d.). Alaska. Retrieved September 27, 2015, from www.math.uaa.alaska.edu/~afkjm/cs351/handouts/pda.pdf

Pushdown Automata - The Stanford University InfoLab. (n.d.). Retrieved September 27, 2015, from infolab.stanford.edu/~ullman/ialc/spr10/slides/pda1.pdf

Pushdown Automata. (2003, October 14). Retrieved September 27, 2015

Pushdown automaton. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Pushdown_automaton

The School of Electrical Engineering and Computer Science. (n.d.). Retrieved from Washington State University: www.eecs.wsu.edu/~ananth/CptS317/Lectures/PDA.pdf