1 PDA conversions

1.1 CFG to PDA

Converting a CFG grammar G to a PDA is done in the following steps:

1. The PDA will only have a single state q
2. For each variable A, $\delta(q, \varepsilon, A) = \{(q, \alpha) \mid A \to \alpha \text{ is in } G\}$
3. For each terminal a, $\delta(q, a, a) = \{(q, \varepsilon)\}$

1.2 PDA to grammar

Our grammar will mostly have variables $[pXq]$, that represent changing from state p to q while popping X from the stack. Important that $[pXq]$ is a single variable.

1. For all states p, introduce $S \to [q_0 Z_0 p]$,
2. For each transition $\delta(q, a, X)$ that contains $(r, Y_1 Y_2 ... Y_k)$, introduce $[qXr_k] \to a[rY_1r_1][r_1Y_2r_2]...[r_{k-1}Y_kr_k]$

1.3 Exercise

Consider the following automaton:

$$P = (\{p.q\}, \{0, 1\}, \{X, Z_0\}, \delta, q, Z_0, \emptyset)$$

with transitions:

$$\begin{aligned}
\delta(q, 1, Z_0) &= \{(q, XZ_0)\}, \\
\delta(q, 1, X) &= \{(q, XX)\}, \\
\delta(q, 0, X) &= \{(p, X)\}, \\
\delta(q, \varepsilon, X) &= \{(q, \varepsilon)\}, \\
\delta(p, 1, X) &= \{(p, \varepsilon)\}, \\
\delta(p, 0, Z_0) &= \{(q, Z_0)\}
\end{aligned}$$

Transform it to a grammar.