A2.1: PDA (3P)

Gesucht war ein deterministischer PDA (DPDA) über $\{a, b, c\}$, der genau die Wörter mit

$$|w|_a \;=\; 2 \cdot |w|_c$$

akzeptiert. **Idee:** Wir halten die Differenz $D = |a| - 2|c|$. Zustand `qPlus` steht für $D \geq 0$ (Stack enthält D-viele X), `qMinus` für $D < 0$ (Stack enthält $-D$-viele Y). Buchstabe b verändert nichts. Akzeptiert wird am Eingabeende genau dann, wenn der Stack nur noch aus dem Bodensymbol besteht ($D = 0$).

Formale Skizze. DPDA $P = (Q, \Sigma, \Gamma, \delta, q_{\text{start}}, Z_0, F)$ mit

$$Q = \{\texttt{qStart}, \texttt{qPlus}, \texttt{qPop1}, \texttt{qMinus}, \texttt{qAfter}, \texttt{qf}\},$$
$$\Sigma = \{a, b, c, \$\}, \quad \Gamma = \{Z_0, X, Y\}, \quad F = \{\texttt{qf}\},$$

und Übergängen wie in deiner Zeichnung (Endmarker-Variante: \$ wird nur in `qPlus` bei Top Z_0 akzeptiert).

Abarbeitung zweier Beispiele.

- *bcaba*: akzeptiert ($|a| = 2, |c| = 1$).
- *bccac*: verworfen ($|a| = 1, |c| = 3 \;\Rightarrow 1 \neq 2 \cdot 3$).

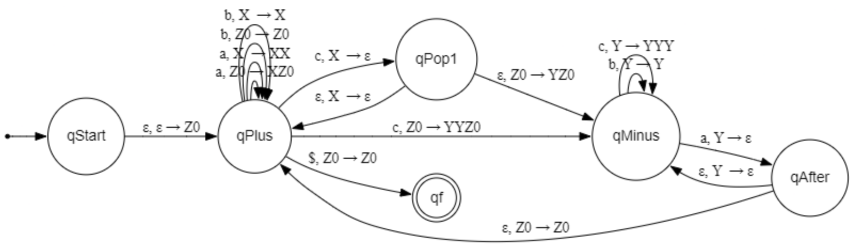

Abbildung 1: A2.1: DPDA für $|w|_a = 2|w|_c$.

A2.2: Akzeptierte Sprache (2P)

Determinismus? Nein. In $\mathtt{q3}$ mit Eingabesymbol d und Top A gibt es zwei Übergänge

$$\delta(\mathtt{q3}, d, A) = (\mathtt{q3}, \varepsilon) \quad \text{und} \quad \delta(\mathtt{q3}, d, A) = (\mathtt{q3}, AA),$$

also nicht deterministisch.

7-Tupel. $P = (Q, \Sigma, \Gamma, \delta, q_0, \perp, F)$ mit

$Q = \{q_0, q_1, q_2, q_3, q_4\}, \quad \Sigma = \{a, b, c, d\}, \quad \Gamma = \{A, B, \perp\},$
q_0 Start, $\perp$ Bodensymbol, $F = \{q_4\}$,
δ wie in der Aufgabenliste (siehe Grafik unten).

Akzeptierte Sprache. Schreibe $w = a^n b^m c^m d^t$; dann gilt

$$L = \{\, a^n b^m c^m d^t \mid n \geq 1,\ m \geq 1,\ t \geq n,\ t \equiv n \pmod 2 \,\}.$$

Begründung: a's pushen A, b's pushen B, c's poppen genau die B ($m = k$), und die d's verändern die Anzahl der A um ± 1 pro Symbol; akzeptiert wird nur, wenn am Ende keine A mehr übrig sind ($t \geq n$ und gleiche Parität).

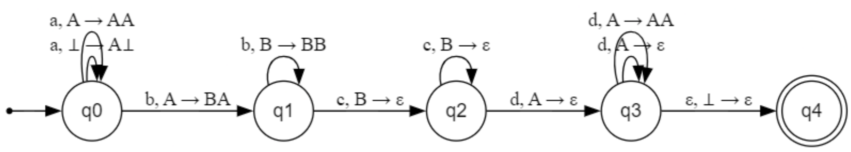

Abbildung 2: A2.2: Der (nichtdeterministische) PDA.

A2.3: Kontextfreie Sprache (2P)

Gegeben ist die Teil-Grammatik:

`Statement` $\rightarrow$ `̈if Condition Statement` | `̈if Condition Statement ̈else Statement`.

Sie erzeugt die Sprache aller *verschachtelten if-(else)-Anweisungen*.

Mehrdeutigkeit. Ja, klassische *dangling-else*-Mehrdeutigkeit: `if C1 if C2 A else A` hat zwei Parsebäume (`else` bindet an das innere oder das äußere `if`).

Unzweideutige Variante (optional). Trennung gematchter und ungematchter Anweisungen:

```
S -> M | U
M -> "if" Condition M "else" M | A      // A: atomare Anweisung
U -> "if" Condition S
```

A2.4: Kontextfreie Grammatik (3P)

Gesucht war eine CFG und ein PDA für

$$L = \{\, a^i b^j c^k \mid i = j \;\vee\; j = k \,\}.$$

Grammatik

```
S -> X C | A Y
X -> a X b |          // i = j
C -> c C |
A -> a A |
Y -> b Y c |          // j = k
```

Diese Grammatik ist *mehrdeutig*, denn Wörter mit $i = j = k$ (z. B. `abc`) lassen sich sowohl über XC als auch über AY ableiten.

PDA (Skizze)

Nichtdeterministische Verzweigung in zwei Zweige:

- Zweig $i = j$: zähle a mit X, poppe bei b, danach beliebige c.
- Zweig $j = k$: ignoriere führende a, zähle b mit Y, poppe bei c.

Akzeptiert wird am Ende (z. B. via Endmarker `$`) genau in einem der Zweige.

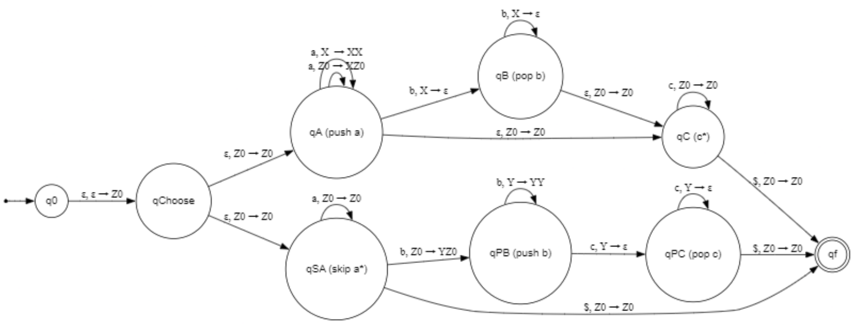

Abbildung 3: A2.4: NPDA für $a^i b^j c^k$ mit i=j oder j=k.