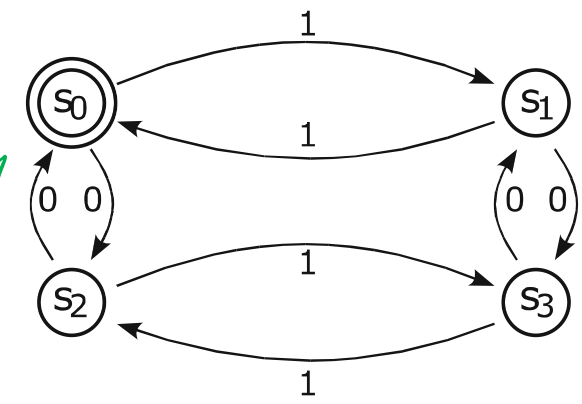
# 2.3 Zustandsautomaten

Wiederholung: $Z = (\mathcal{S}, \mathcal{X}, \mathcal{T}, s_0, \mathcal{F})$ mit

- $\mathcal{S}$ : Menge von Zuständen z.B. $\{s_0, s_1, s_2; s_3\}$
- $\mathcal{X}$ : Eingabealphabet z.B. $\{0, 1\}$
- $\mathcal{T}$ : Transitionsfunktionen $t(s^-, x_i) = s^+$
- $s_0$: Startzustand ($s_0 \in \mathcal{S}$)
- $\mathcal{F}$ : Menge von zulässigen Endzuständen ($\mathcal{F} \subseteq \mathcal{S}$)




a1) Woran erkannt man einen endlichen Zustandsautomaten?

Endliche Menge von Zuständen S

a2) Woran erkennt man einen deterministischen Zustandsautomaten?

Der Folgezustand ist immer eindeutig definiert durch:

- aktuellen Zustand und
- Eingabesymbol

Sie entwerfen einen Getränkeautomaten als Zustandsautomat. Der Automat akzeptiert Münzen mit 50ct und 1€ (andere Münzarten können Sie vernachlässigen). Ein Getränk kostet 1,50€. Nach der Entnahme eines Getränks ist der Automat wieder zur Geldaufnahme bereit. Wird zu viel Geld eingeworfen, bricht der Automat den Vorgang ab und gibt anschließend das eingeworfene Geld zurück. Zudem kann der Vorgang jederzeit manuell über eine Rückgabetaste abgebrochen werden.

b1) Definieren Sie die nötige Zustandsmenge $\mathcal{S}$ und das Eingabealphabet $\mathcal{X}$.

**Zustände:**

$S_0$: Bereit (0,-€ eingeworfen)

$S_1$: Bereit (0,50€ ")

$S_2$: Bereit (1,-€ ")

$S_3$: Bereit (1,50€, ")

$S_4$: Abgebrochen

**Eingabesymbole:**

$x_{50ct}$: Einwurf 50ct

$x_{1EUR}$: " 1,- €

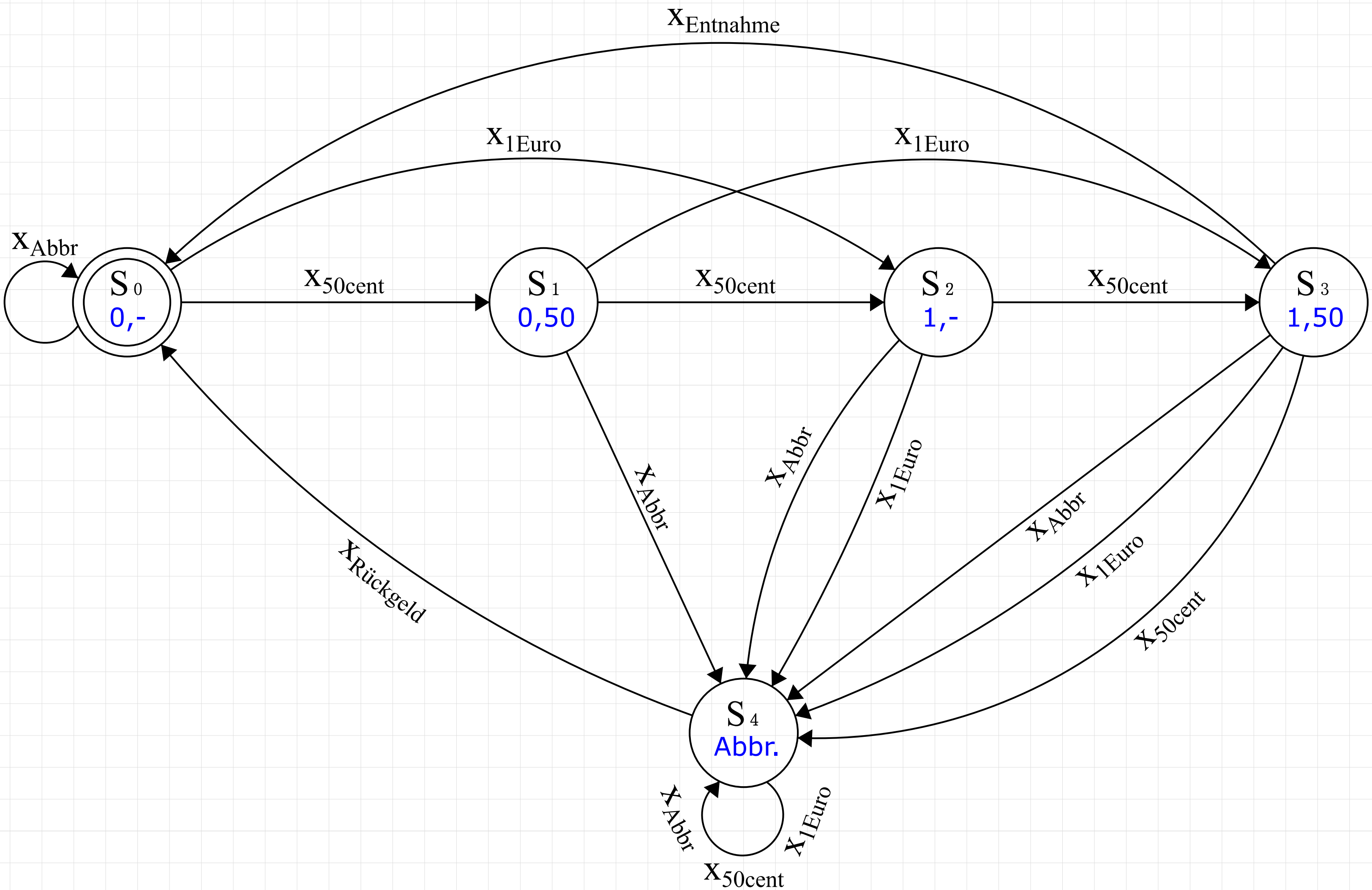
$x_{Abbr}$: Abbruchtaste drücken

$x_{Entnahme}$: Getränk entnehmen

$x_{Rückgeld}$: Geld zurückgenommen

# 2.3 Zustandsautomaten

b2) Zeichnen Sie das zugehörige Zustandsdiagramm mit allen mögl. Transitionen.



## Hinweis:

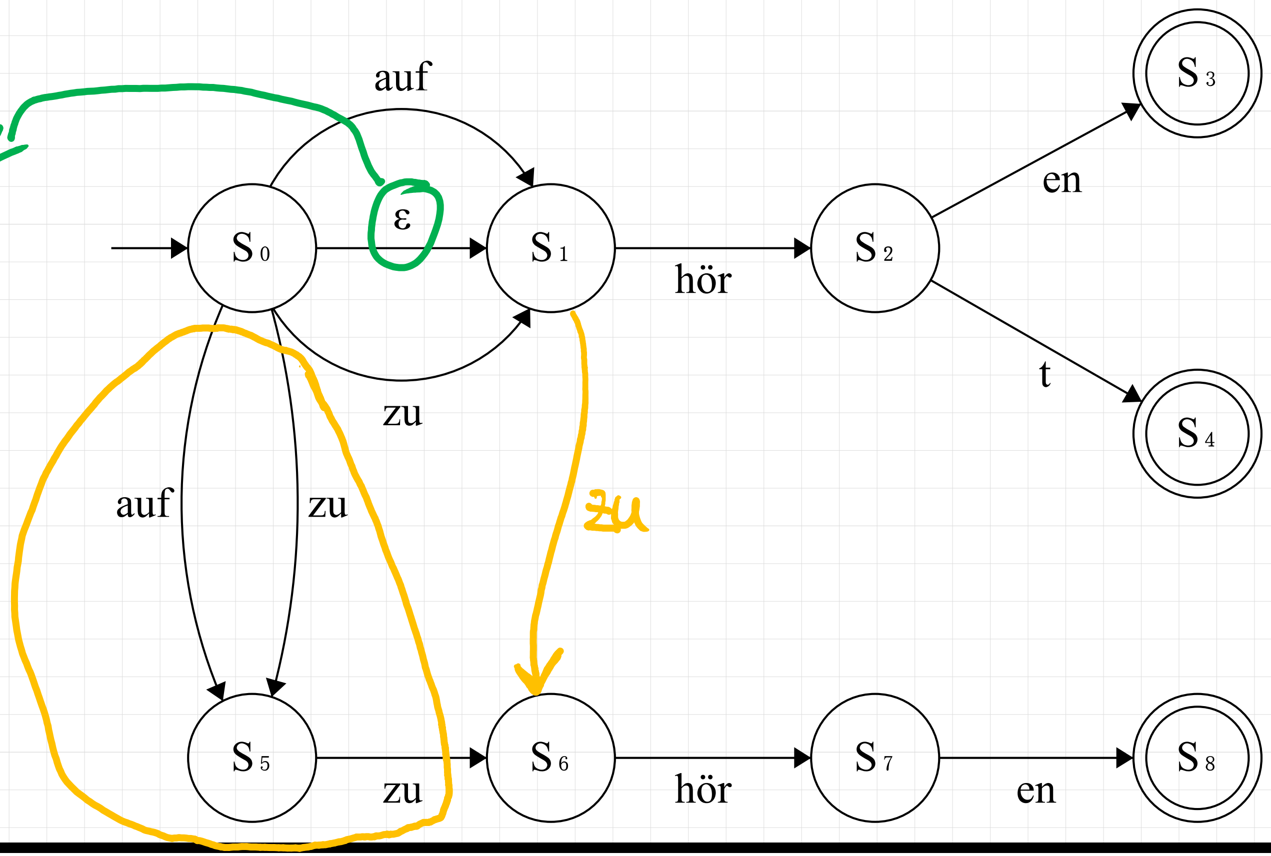
- andere Lösungen sind möglich
- wenn wir nicht an jedem Zustand für jedes Symbol eine Transition verwenden, meinen wir damit implizit:

  falls es nicht weiter geht (mit Symbol x im Zustand S), so wird das Wort nicht akzeptiert (d.h. eine gedachte Transition auf einen Fehlerzustand)

c) Zeichnen Sie einen endlichen Automaten, für den Phoneme als Eingabesymbole fungieren und der ausschließlich die folgenden Wörter (= Symbolfolgen) akzeptiert:

```
hör-en        zu-hör-en   auf-hör-en   auf-zu-hör-en
zu-zu-hör-en  hör-t       zu-hör-t     auf-hör-t
```

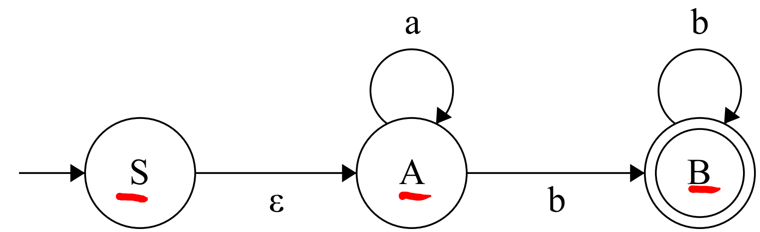
ε: leere Eingabe bzw. spontane Transition




Alternative: streiche $S_5$

Gegeben ist der folgende endliche Automat A:

d1) Beschreiben Sie informell die von A erkannte Menge von Symbolfolgen.




→ alle Wörter mit beliebig vielen "a" vor mind. einem "b"

als regulärer Ausdruck: $r = a^* b b^*$

d2) Geben Sie eine Grammatik an, die die von A verarbeitbaren Wörter beschreibt

1. Markiere Zustände mit Nichtterminalsymbolen $V = \{S, A, B\}$

2. Übertrage jede Transition in eine Produktionsregel

$P = \{S \rightarrow A, A \rightarrow aA, A \rightarrow bB, B \rightarrow bB, B \rightarrow \varepsilon\}$

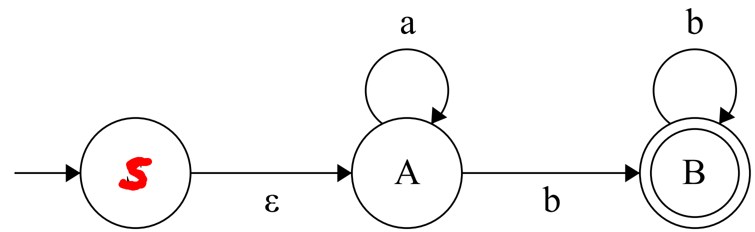
3. Für jeden Endzustand: erstelle Regel mit leerem Wort rechts

$\Rightarrow G = (V, T, P, S)$ mit $T = \{a, b\}$

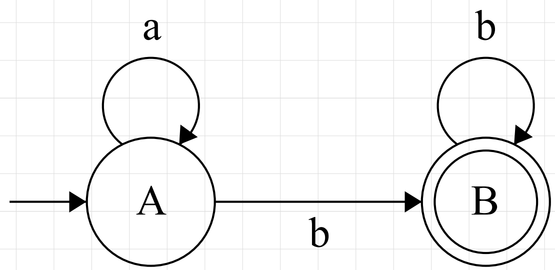
# 2.3 Zustandsautomaten

Gegeben ist der folgende endliche Automat A:

d3) Zeichnen Sie eine Zustandsautomaten, der die gleiche Sprache wie A akzeptiert, aber nur aus 2 Zuständen besteht.




- S entfällt
- A neuer Startzustand




d4) Wie ändert sich dadurch die zugehörige Grammatik?

$V = \{A, B\}$ $\quad S = A$

$G = (V, T, P, S)$

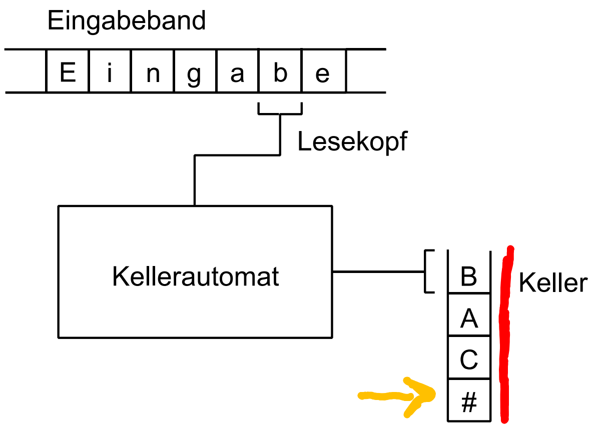
$P = \{A \rightarrow aA,\ A \rightarrow bB,\ B \rightarrow bB,\ B \rightarrow \varepsilon\}$

Wiederholung: Kellerautomaten (*Push-Down Automata*)

K = ($\mathcal{S}$, $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{T}$, $s_0$, $y_0$, $\mathcal{F}$) mit zusätzlichen Elementen…

- $\mathcal{Y}$: Zulässiges Alphabet für die Symbole im Stack
- $y_0$: Startsymbol für den Stack

First-In, Last-Out

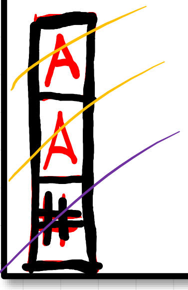


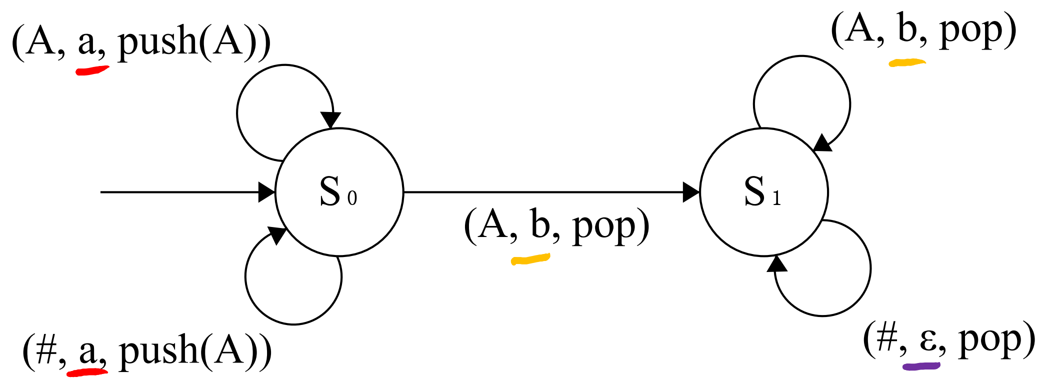

Aktionen:
- push(x): lege x auf den Stack
- pop(): lese (und entferne) oberstes Stack-Element
- ε: keine Aktion

Übergangsfkt.: $\delta = (X \times S \times Y) \to (S \times Y^*)$

Bildquelle: Martin Hampl, https://de.wikipedia.org/wiki/Kellerautomat#/media/File:Kellerautomat.svg (abgerufen 21.11.2016) e1) Beschreiben Sie die Sprache, die folgender Kellerautomat erkennt:

aabb




1.Schritt

Letzter Schritt

$L(a^n b^n)\ , \quad n > 0$

e2) Geben Sie eine kontextfreie Grammatik an, die die Sprache von e1) generiert.

$G = \langle \{S\}, \{a,b\}, \{S \rightarrow aSb, S \rightarrow ab\}, S \rangle$

V T P S

f) Geben Sie einen Kellerautomaten K an, der die Sprache L($a^n$ b $c^n$) akzeptiert. $n > 0$ $n \geq 0$

Es sei vorgegeben: $\mathcal{Y} = \{A, \#\}$ $y_0 = \#$ $\mathcal{F} = \{\ \}$ d.h.: bei leerem Stack ist das Ende erreicht (und ein Wort „akzeptiert")