

Formale Sprachen am Beispiel von Passwörtern

February 1, 2025

1 Einleitung

Passwörter sind ein ausgezeichnetes Beispiel, um die Konzepte von formalen Sprachen und Grammatiken zu vermitteln. Passwörter folgen klar definierten Regeln (oder einer "Grammatik"), die festlegen, welche Kombinationen von Zeichen zulässig sind. In diesem Skript wird gezeigt, wie diese Regeln in einer regulären Grammatik dargestellt werden können, wie Schüler Passwörter analysieren können und wie diese Konzepte in der Praxis angewendet werden.

2 Darstellung der Passwortregeln als Grammatik

2.1 Passwortregeln

Angenommen, ein Passwort muss die folgenden Anforderungen erfüllen:

1. Es muss mindestens 8 Zeichen lang sein.
2. Es muss mindestens einen Buchstaben enthalten.
3. Es muss mindestens eine Zahl enthalten.
4. Es muss mindestens ein Sonderzeichen enthalten (z. B. !, @, #).

2.2 Reguläre Grammatik

Diese Regeln können mithilfe einer regulären Grammatik beschrieben werden. Die Grammatik besteht aus folgenden Elementen:

Nichtterminale:

- **Passwort** – Startsymbol der Grammatik
- **Zeichenkette** – Eine Folge von Zeichen
- **Zeichen** – Ein einzelnes Zeichen (Buchstabe, Zahl oder Sonderzeichen)

Terminale:

- Buchstaben: $a\{z, A\{Z$
- Zahlen: $0\{9$
- Sonderzeichen: $!, @, \#, \text{etc.}$

2.3 Grammatikregeln

1. $\text{Passwort} \rightarrow \text{Zeichenkette}$
2. $\text{Zeichenkette} \rightarrow \text{Zeichen Zeichenkette} \mid \varepsilon$
3. $\text{Zeichen} \rightarrow \text{Buchstabe} \mid \text{Zahl} \mid \text{Sonderzeichen}$
4. $\text{Buchstabe} \rightarrow a \mid b \mid c \mid \dots \mid z \mid A \mid B \mid C \mid \dots \mid Z$
5. $\text{Zahl} \rightarrow 0 \mid 1 \mid 2 \mid \dots \mid 9$
6. $\text{Sonderzeichen} \rightarrow ! \mid @ \mid \# \mid \$ \mid \% \mid \wedge \mid \& \mid * \mid (\mid) \mid \dots$

Hierbei steht ε für die leere Zeichenkette (Ende der Zeichenkette).

2.4 Konkrete Ableitung eines Passworts

Nehmen wir das Passwort `A1b@2023` und leiten es nach den Regeln ab:

```
Passwort → Zeichenkette
Zeichenkette → Zeichen Zeichenkette
Zeichen → A (Buchstabe)
Zeichenkette → Zeichen Zeichenkette
Zeichen → 1 (Zahl)
...
Zeichen → 3 (Zahl)
Zeichenkette → (Ende)
```

Diese Ableitung zeigt, dass `A1b@2023` gemäß der Grammatik aufgebaut ist und somit ein gültiges Passwort darstellt.

3 Praxisübung: Passwortanalyse

3.1 Beispielhafte Passwörter zur Analyse

Passwort	Analyse	Ergebnis
Hallo123!	8 Zeichen, Buchstaben, Zahl, ‘!	✓Gültig
Passwort	Nur Buchstaben	✗Ungültig
12345678	Nur Zahlen	✗Ungültig
Abc@2023	8 Zeichen, Buchstaben, Zahl, ‘@’	✓Gültig
Hallo12	Nur 7 Zeichen	✗Ungültig

4 Computersysteme und Passwortvalidierung

Computersysteme prüfen diese Regeln automatisiert. Hier ein vereinfachtes Python-Beispiel:

```
import re

def validiere_passwort(password):
    # Regulärer Ausdruck zur Prüfung der Regeln
    regel = r'^(?=.*[A-Za-z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}$'
    return bool(re.match(regel, password))

# Testbeispiele
print(validiere_passwort("Hallo123!")) #
    True
print(validiere_passwort("Passwort")) #
    False
print(validiere_passwort("12345678")) #
    False
```

5 Fazit

Die Darstellung von Passwortregeln als Grammatik verbindet Theorie und Praxis. Sie zeigt Schülern, wie formale Sprachen ihren Alltag beeinflussen.