

# Algorithmen und Datenstrukturen

---

## Aufgabe 1 (AGS 2.2.39)

Sei  $\mathcal{E} = (\{S, B\}, \{b\}, S, R)$  mit  $R = \{ S ::= \widehat{B}b, B ::= Sb \}$  eine EBNF-Definition. Berechnen Sie die syntaktischen Kategorien  $W(\mathcal{E}, S)$  und  $W(\mathcal{E}, B)$  mit Hilfe der Fixpunktsemantik. Gehen Sie dazu in den folgenden Schritten vor:

- Dokumentieren Sie mindestens 5 Iterationsschritte,
- und schreiben Sie in Mengenschreibweise die Sprachen  $W(\mathcal{E}, S)$  und  $W(\mathcal{E}, B)$  auf.
- Zeigen Sie mithilfe der Semantik von EBNF-Termen, dass  $\rho: V \rightarrow \mathcal{P}(\Sigma^*)$  mit

$$\rho(S) = \rho(A) = \{(ba)^n b \mid n \in \mathbb{N}\}$$

die EBNF-Regel  $S ::= \widehat{ba}A$  erfüllt.

## Aufgabe 2 (AGS 3.1.2)

Schreiben Sie jeweils ein Programm, das

- einen Namen einliest und eine Begrüßung ausgibt,
- für eine Eingabe  $n$  eine Multiplikationstabelle für die Zahlen von 1 bis  $n$  ausgibt, und
- die Primzahlen von 1 bis 1000 berechnet und ausgibt.

## Aufgabe 3 (AGS 3.1.6)

Schreiben Sie für die Berechnung der Fakultätsfunktion  $n \mapsto n! = n \cdot (n-1) \cdot \dots \cdot 2 \cdot 1$ ,  $n \in \mathbb{N}$ , ein C-Programm, welches eine natürliche Zahl als Eingabe fordert und den zugeordneten Funktionswert ausgibt.

## Zusatzaufgabe 1 (AGS 2.2.40)

- Geben Sie die Mengen  $V$  und  $R$  der EBNF-Definition  $\mathcal{E} = (V, \{a, b, c\}, S, R)$  an, sodass

$$W(\mathcal{E}, S) = \{a^{i+j}b^{j+k+l}ac^{2l} \mid i, j, k, l \geq 0\}.$$

- Sei  $\mathcal{E}' = (\{S, A\}, \{a, b\}, S, R')$  mit  $R' = \{ S ::= aAb, A ::= ([S] \mid b) \}$ . Berechnen Sie die syntaktischen Kategorien  $W(\mathcal{E}, S)$  und  $W(\mathcal{E}, A)$  mit Hilfe der Fixpunktsemantik. Gehen Sie dazu in den folgenden Schritten vor:
  - Dokumentieren Sie 5 Iterationsschritte.
  - Schreiben Sie in Mengenschreibweise die Sprachen  $W(\mathcal{E}', S)$  und  $W(\mathcal{E}', A)$  auf.

## Zusatzaufgabe 2 (AGS 3.1.17)

Schreiben Sie ein Programm, das den Nutzer eine zufällig bestimmte Zahl von 1 bis  $n$  raten lässt. Es soll nur ausgegeben werden, ob die geratene Zahl größer, kleiner, oder gleich der gesuchten Zahl ist.

*Zusatz<sup>2</sup>aufgabe:* Bewerten Sie den Nutzer im Vergleich zur optimalen Ratestrategie!

### **Zusatzaufgabe 3 (AGS 3.1.3)**

Ein ganzzahliger Geldbetrag in Euro soll in die Grundeinheiten 500, 200, 100, 50, 20, 10, 5, 2 und 1 zerlegt werden.

Schreiben Sie ein C-Programm, welches einen Geldbetrag als Eingabe fordert, dann die Vielfachheiten der oben genannten Grundeinheiten berechnet und das Ergebnis auf dem Bildschirm ausgibt. Entwickeln Sie einen solchen Algorithmus, der ein Minimum an Geldscheinen und Münzen erzeugt. Geben Sie zunächst den Algorithmus in Pseudocode an.