

## Übungsaufgaben (für Kapitel 1 bis 3)

### Aufgabe 1: Multiplikation

Beschreiben Sie die Multiplikation zweier natürlicher Zahlen.

- a) Verwenden Sie eine iterierte Addition.
- b) Beschreiben Sie den zeichenweise arbeitenden Algorithmus, den Sie aus der Grundschule kennen.
- c) Nehmen Sie an, Sie hätten als Operation die Bildung des Quadrats einer Zahl zur Verfügung, d.h., es gibt den Operator  $\text{square}(x)$ , der zu einer natürlichen Zahl  $x$  das Quadrat  $x^2$  liefert. Wie kann man dann die Multiplikation durchführen?

## Aufgabe 2: *Berechnung von div und mod*

Der Euklidische Algorithmus für den größten gemeinsamen Teiler ggT benutzt die Operation mod, die zu zwei natürlichen Zahlen  $a$  und  $b$  den Rest liefert, der bei der ganzzahligen Division  $a \text{ div } b$  von  $a$  durch  $b$  übrig bleibt.

Es gilt:  $a = b \cdot (a \text{ div } b) + a \text{ mod } b$  für  $a \geq 0$  und  $b > 0$ .

Geben Sie zwei Algorithmen an, die  $a \text{ div } b$  und  $a \text{ mod } b$  gleichzeitig berechnen, und zwar

- a) einmal zeichenweise (z.B. als array dargestellt) und
- b) einmal ohne die Darstellung mit Ziffern zu verwenden.

Prüfen Sie Ihre Algorithmen mit einigen Ablaufprotokollen.

Aufgabe 3: *Ermitteln der Größe des Durchschnitts*

Gegeben seien zwei Feldvariablen A und B

type Zahlenfeld is array [1..n] of integer;

Zahlenfeld A, B;

Jede der Variablen A und B stellt eine Teilmenge der ganzen Zahlen dar und wir nehmen an, dass keine Zahl in A (bzw. in B) mehrfach vorkommt:

$$\forall i, j \in \{1, 2, \dots, n\} \text{ and } (i \neq j) \Rightarrow A[i] \neq A[j].$$

Schreiben Sie ein Programm, welches n und die Felder A und B einliest (Sie können z.B.  $n \leq 10000$  annehmen) und

1. feststellt, dass keine Zahl in A und in B mehrfach vorkommt, und in diesem Fall
2. ermittelt, wie viele Zahlen sowohl in A als auch in B vorkommen (Größe des Durchschnitts der durch A und B definierten Teilmengen von **Z**).

#### Aufgabe 4: Zeichenverschiebung und quadratische Codierung

Es sei  $\text{Pos}: \mathbf{A} \rightarrow \{0, 1, 2, \dots, 127\}$  die Funktion, die jedem Zeichen seine Nummer im ASCII-Code zuordnet (siehe 1.4, z.B. Folien 52 und 57).

Es sei  $\text{Val}: \{0, 1, 2, \dots, 127\} \rightarrow \mathbf{A}$  die Funktion, die jeder Zahl zwischen 0 und 127 das zugehörige Zeichen bzgl. des ASCII-Codes zuordnet.

a) Schreiben Sie einen Algorithmus, der zunächst eine Zahl  $k$  und danach eine Folge von Zeichen (das Ende der Folge sei durch ein von Ihnen zu bestimmendes Zeichen festgelegt) einliest und die um  $k$  Stellen im ASCII-Code zyklisch verschobene Folge dieser Zeichen ausgibt.

b) Wir wollen die Zeichen durch folgende Vorschrift verschlüsseln: Wenn das Zeichen  $\alpha$  die Nummer  $\text{Pos}(\alpha) = i$  besitzt, dann berechnen wir  $j = i^2 \bmod 128$  und ersetzen  $\alpha$  durch das Zeichen  $\beta = \text{Val}(j)$ .

Beispiel:  $\alpha = 'U'$ ,  $\text{Pos}(U) = i = 85$ ,  $j = 85^2 \bmod 128 = 7225 \bmod 128 = 57$ ,  $\beta = \text{Val}(57) = '9'$ .

Schreiben Sie einen Algorithmus, der diese Verschlüsselung vornimmt, der also die Abbildung  $\alpha \rightarrow \beta = \text{Val}((\text{Pos}(\alpha))^2 \bmod 128)$  berechnet.

Ist diese Abbildung injektiv?

Diskutieren Sie Vor- und Nachteile dieser Abbildung.

### Aufgabe 5: Datentypdeklarationen

Beschreiben Sie folgende Daten mit Hilfe der in Kapitel 1 und 2 vorgestellten Datentypen und Konstruktoren:

- a) Liste der Nummern der Großbuchstaben im ASCII\_Code.
- b) Menge der Vornamen aller Freunde und Freundinnen.
- c) Zuordnung der Geburtsjahre zu jedem/r Freund/Freundin.
- d) Datentyp für die Wochentage und Datentyp für sieben selbst zu definierende Wetterlagen (kühl, regnerisch, sonnig, ...).
- e) Zuordnung von Wochentag und Wetterlage zu der bei dieser Situation geeigneten Kleidung.
- f) Jede Teilmenge einer angeordneten Menge  $M$  mit  $n$  Elementen kann man durch einen  $n$ -stelligen Booleschen Vektor beschreiben, dessen  $i$ -te Komponente genau dann true ist, wenn das  $i$ -te Element in der Teilmenge liegt. Definieren Sie auf diese Weise den Datentyp "Menge der Teilmengen von  $M$ ".

### Aufgabe 6: Terminierung?

Ermitteln Sie, für welche Eingabewerte folgende Programme terminieren (= nach endlich vielen Schritten anhalten) und welche Funktion sie realisieren.

program P1 is

integer k, m;

begin read m; k := 1;

while m > k do k := k+2; m := m+1 od;

write k

end;

program P2 is

integer k, m;

begin read m; k := -m;

while m > 0 do k:=k+2; m:= k od;

write k

end;

### Aufgabe 7: Terminierung?

Ermitteln Sie, für welche Eingabewerte folgendes Programm terminiert (zu "Pos" siehe Aufgabe 4).

program unbekannt is

integer k; character X; array [0..3] of character DB;

begin

for k:=0 to 3 do DB[k] := 'Z' od;

read X;

while X ≠ 'Z' do

        k := Pos(X);

while DB[k] ≠ 'Z' do k := (k+1) mod 4 od;

        DB [k] := X

od;

for k:=0 to 3 do write DB[k] od

end;

### Aufgabe 8: Operationen auf Sprachen

Es sei  $A$  die zweielementige Menge  $A = \{a, b\}$ .

- (1) Bilde zur Menge  $L = \{a, bb\}$  die Mengen  $LL$  und  $LLL$ .
- (2) Skizzieren Sie ein Verfahren, das zu einem Wort  $w \in A^*$  feststellt, ob  $w \in L^*$  gilt oder nicht.
- (3) Sei  $J = \{ab\}$ . Beschreiben Sie die Mengen  $J^*$  und  $A^* \setminus J^*$ .
- (4) Sei  $K = \{b, ab\}$ . Beschreiben Sie die Menge  $(K \cup L)^*$ .
- (5) Beschreiben Sie die Menge  $\{(abb)^n \mid n > 0\}$  mit Hilfe der Sprachen  $J$  und  $K$  und der Sprachoperationen Vereinigung, Differenz, Durchschnitt, Konkatenation und/oder Iteration.
- (6) Sei  $H = \{ba, bb\}$ . Geben Sie einen Algorithmus für das Wortproblem der Menge  $(H \cup J)^*$  an, d.h., geben Sie ein Verfahren an, das zu jedem Wort  $w \in A^*$  feststellt, ob  $w \in (H \cup J)^*$  gilt oder nicht.



Aufgabe 9: *Erzeugte kontextfreie Sprachen*

Es seien  $V = \{S, A\}$  und  $\Sigma = \{0, 1\}$ . Welche Sprachen  $L_i$  werden von folgenden kontextfreien Grammatiken

$G_i = (V, \Sigma, P_i, S)$  erzeugt mit  $i = 1, 2, \dots, 9$  und

$$P_1 = \{ S \rightarrow 0, S \rightarrow 1 \}$$

$$P_2 = \{ S \rightarrow 1, S \rightarrow S, S \rightarrow 0, A \rightarrow 0 \}$$

$$P_3 = \{ S \rightarrow 0S, S \rightarrow 00 \}$$

$$P_4 = \{ S \rightarrow S1, S \rightarrow 00, S \rightarrow 1, A \rightarrow 1, A \rightarrow AS \}$$

$$P_5 = \{ S \rightarrow 0A0, S \rightarrow 1, A \rightarrow 0S0 \}$$

$$P_6 = \{ S \rightarrow 0AS, S \rightarrow 0, A \rightarrow 1A1, A \rightarrow 1 \}$$

$$P_7 = \{ S \rightarrow AA, A \rightarrow 0A0, A \rightarrow 1 \}$$

$$P_8 = \{ S \rightarrow AS, A \rightarrow \varepsilon, A \rightarrow 0S1, S \rightarrow A \}$$

$$P_9 = \{ S \rightarrow SAS, A \rightarrow 10, A \rightarrow 0S1A, S \rightarrow A \}$$

Aufgabe 10: *(Beliebige) Grammatiken konstruieren*

Konstruieren Sie Grammatiken  $G_i = (V_i, \Sigma, P_i, S)$  mit  $\Sigma = \{0, 1\}$ , die folgende Sprachen  $L_i$  erzeugen (hier ist  $w^R$  das gespiegelte Wort  $w$ , d.h., wenn  $w = a_1a_2\dots a_n$  ist, dann ist  $w^R = a_n\dots a_2a_1$ ):

$$L_1 = \Sigma^* \setminus \{\varepsilon\} = \Sigma^+$$

$$L_2 = \{ w \in \Sigma^* \mid w \text{ hat eine gerade Länge} \}$$

$$L_3 = \{ 0^n 1 0^n \mid n > 0 \}$$

$$L_4 = \{ (0^n 1 0^n)^m \mid n > 0, m > 0 \}$$

$$L_5 = \{ ww^R \mid w \in \Sigma^* \}$$

$$L_6 = \{ w^2 \mid w \in \Sigma^* \}$$

$$L_7 = \{ 0^n 1 0^n 1 0^n \mid n > 0 \}$$

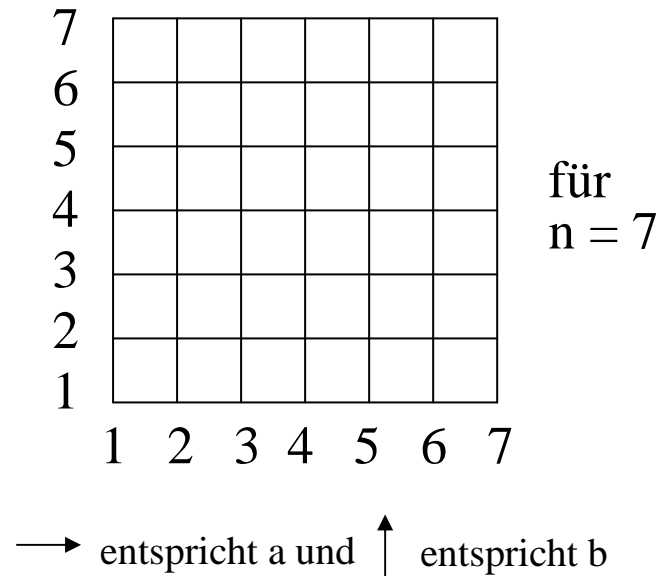
$$L_8 = \{ w \in \Sigma^* \mid w \text{ besitzt gleich viele Zeichen } 0 \text{ und } 1 \}$$

$$L_9 = \{ 0^n \mid \text{Es gibt ein } k \geq 0 \text{ mit } n = 2^k \}$$

### Aufgabe 11: Charakterisierung von "diagonalen Wegen"

Es sei ein  $n \times n$ -Gitter gegeben:

Es sollen genau die Wege beschrieben werden, die vom Punkt (1,1) unten links zum Punkt (n,n) oben rechts führen. Hierfür bedeute das Zeichen a "gehe zum nächsten Punkt rechts" und b bedeute "gehe zum nächsten Punkt nach oben". Das Wort  $abab..ab = (ab)^n$  beschreibt dann einen solchen "diagonalen Weg" ebenso wie das Wort  $a^n b^n$ .



- Beschreiben Sie die Menge aller Wörter, die einen Weg vom Punkt (1,1) zum Punkt (n,n) beschreiben.
- Geben Sie eine Grammatik an, die für alle  $n$  genau alle diese Wörter erzeugt.

## Aufgabe 12: Umwandlung in Syntaxdiagramme

Geben Sie zu folgenden kontextfreien Grammatiken

$G_i = (V, \Sigma, P_i, S)$  mit  $V = \{S, A, B\}$  und  $\Sigma = \{0, 1, 2, 3\}$

möglichst einfache Syntaxdiagramme an:

$$P_1 = \{ S \rightarrow 0SS, S \rightarrow 1 \}$$

$$P_2 = \{ S \rightarrow AA, S \rightarrow 0, A \rightarrow AA, A \rightarrow 1, A \rightarrow AS \}$$

$$P_3 = \{ S \rightarrow 0B1S3, S \rightarrow 0B1S2S3, S \rightarrow 00, B \rightarrow 11 \}$$

$$P_4 = \{ S \rightarrow A0S, S \rightarrow A, A \rightarrow B1A, A \rightarrow B, B \rightarrow 2S2, B \rightarrow 3 \}$$

$$P_5 = \{ S \rightarrow ABS, A \rightarrow 0A1, A \rightarrow \varepsilon, B \rightarrow 0B, B \rightarrow \varepsilon, S \rightarrow 2 \}$$

$$P_6 = \{ S \rightarrow A1B, S \rightarrow B1A, A \rightarrow 0A0, A \rightarrow 1, B \rightarrow 0B, B \rightarrow 0 \}$$