

Wintersemester 2024/25

Übungen zur Vorlesung Automaten und Formale Sprachen

Blatt 8

Termine: 16. Dezember 2024 und 13. Januar 2025

### Primitiv-rekursive und $\mu$ -rekursive Funktionen

#### Aufgabe 1

Konstruieren Sie nach dem Prinzip der primitiven Rekursion eine Funktion, die:

1. die Multiplikation  $\text{mult}(m, n) := m * n$  realisiert
2. die modifizierte Differenz  $\text{diff}(m, n) := m \ominus n := \begin{cases} m - n & \text{falls } m \geq n \\ 0 & \text{sonst} \end{cases}$  realisiert
3. die absolute Differenz  $\text{absdiff}(m, n) := |m - n| := \begin{cases} m - n & \text{falls } m \geq n \\ n - m & \text{sonst} \end{cases}$  realisiert
4. feststellt, ob eine Zahl ungerade ist oder nicht:  $\text{odd}(n) := \begin{cases} 1 & \text{falls } n \text{ ungerade} \\ 0 & \text{sonst} \end{cases}$
5. das Minimum zweier Zahlen berechnet:  $\text{min}(m, n) := \begin{cases} m & \text{falls } m \leq n \\ n & \text{sonst} \end{cases}$

Hinweise:

- Zum besseren Verständnis und zum Testen ist es hilfreich, die Funktionen in Java zu programmieren. Ausgewählte Funktionen, die Sie auch nutzen können, finden sich in der Datei `Blatt_8_Funktionen.zip`. Beachten Sie, dass Sie keine Kontrollstrukturen verwenden dürfen – außer eine äußere if-Abfrage zur Unterscheidung zwischen dem Basisfall und dem Rekursionsfall.
- Beachten Sie, dass es genau einen Basisfall ( $n = 0$ ) gibt und in dem Rekursionsfall das Argument um 1 erniedrigt wird.
- Die 3. und 5. Teilaufgabe lassen sich am einfachsten durch Komposition (ohne Rekursion) lösen.

#### Aufgabe 2

Programmieren Sie in einer Programmiersprache Ihrer Wahl wie etwa Java oder C++ die Ackermann-Funktion  $\text{ack}(n, m)$ . Rufen Sie die Funktion mit ausgewählten Werten (z.B.  $(3, 2)$ ,  $(3, 3)$  und  $(4, 1)$ ) auf und verfolgen Sie die rekursiven Aufrufe durch entsprechende Kontrollausgaben.

#### Aufgabe 3

Bestimmen Sie für die folgenden Funktionen  $f$  jeweils  $\mu f$ :

1.  $f(m, n) = (9 \ominus m^2) \cdot (n^2 \ominus 4)$
2.  $f(k, m, n) = (m \ominus k) + (n \ominus k)$
3.  $f(m, n) = \text{absdiff}(m, n)$ .

#### Aufgabe 4

Erstellen Sie jeweils eine Funktion  $f : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ , sodass

1.  $\mu f(n) = 3 * n$

2.  $\mu f(n) = \begin{cases} 0 & \text{für } 0 \leq n \leq 3 \\ \perp & \text{sonst} \end{cases}$

gilt.

#### Hinweis zur Abgabe

Erstellen Sie ein pdf-Dokument mit den Lösungen zu den Aufgaben. Für die Aufgabe 2 kopieren Sie den Quellcode Ihrer Implementierung der Ackermann-Funktion in das Abgabedokument. Legen Sie das Dokument mit dem Namen **Blatt8\_<Nachname1>\_<Nachname2>.pdf** bis spätestens **Freitag, 17. Januar 2025**, in dem Abgabeordner **Blatt 8** ab.