Grundbegriffe der Informatik

Tutorium 1 - 10. Sitzung

Dennis Felsing

dennis.felsing@student.kit.edu
http://www.stud.uni-karlsruhe.de/~ubcqr/2010w/tut_gbi/

2011-01-10

Karlsruhe Institute of Technology

Überblick

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n$
- $n^2 + n^3$
- $3 logn$
- $5n \cdot 3n$
- n^n
- $12\sqrt{5n}$
- 37^{14}
- $n^2 logn + n$

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n$
- $n^2 + n^3$
- $3 log n$
- $5n \cdot 3n$
- n^n
- $12\sqrt{5n}$
- $37^{14} \in \Theta(1)$ (1)
- $n^2 log n + n$

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n$
- $n^2 + n^3$
- $3 log n \in \Theta(log n)$ (2)
- $5n \cdot 3n$
- n^n
- $12\sqrt{5n}$
- $37^{14} \in \Theta(1)$ (1)
- $n^2 log n + n$

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n$
- $n^2 + n^3$
- $3logn \in \Theta(logn)$ (2)
- $5n \cdot 3n$
- n^n
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n$

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n \in \Theta(n)$ (4)
- $n^2 + n^3$
- $3 logn \in \Theta(logn)$ (2)
- $5n \cdot 3n$
- n^n
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n$

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n \in \Theta(n)$ (4)
- $n^2 + n^3$
- $3 logn \in \Theta(logn)$ (2)
- $5n \cdot 3n \in \Theta(n^2)$ (5)
- n^n
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n$

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n \in \Theta(n)$ (4)
- $n^2 + n^3$
- $3logn \in \Theta(logn)$ (2)
- $5n \cdot 3n \in \Theta(n^2)$ (5)
- n^n
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n \in \Theta(n^2 logn)$ (6)

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n!$
- $2 + 3n \in \Theta(n)$ (4)
- $n^2 + n^3 \in \Theta(n^3)$ (7)
- $3logn \in \Theta(logn)$ (2)
- $5n \cdot 3n \in \Theta(n^2)$ (5)
- n^n
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n \in \Theta(n^2 logn)$ (6)

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n! \in \Theta(n!)$ (8)
- $2 + 3n \in \Theta(n)$ (4)
- $n^2 + n^3 \in \Theta(n^3)$ (7)
- $3 logn \in \Theta(logn)$ (2)
- $5n \cdot 3n \in \Theta(n^2)$ (5)
- n^n
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n \in \Theta(n^2 logn)$ (6)

Wiederholung

Ordne die folgenden Funktionen nach ihrem asymptotischen Wachstum:

- $n! \in \Theta(n!)$ (8)
- $2 + 3n \in \Theta(n)$ (4)
- $n^2 + n^3 \in \Theta(n^3)$ (7)
- $3 logn \in \Theta(logn)$ (2)
- $5n \cdot 3n \in \Theta(n^2)$ (5)
- $n^n \in \Theta(n^n)$ (9)
- $12\sqrt{5n} \in \Theta(\sqrt{n})$ (3)
- $37^{14} \in \Theta(1)$ (1)
- $n^2 logn + n \in \Theta(n^2 logn)$ (6)

Mastertheorem

Wir betrachten Gleichungen dieser Form:

$$T(n) = a \cdot T(\tfrac{n}{b}) + f(n)$$

Mastertheorem

Wir betrachten Gleichungen dieser Form:

$$T(n) = a \cdot T(\tfrac{n}{b}) + f(n)$$

Drei Fälle werden unterschieden:

1. $f(n) \in O(n^{\log_b a - \varepsilon})$ für ein $\varepsilon > 0$
   $\Rightarrow T(n) \in \Theta(n^{\log_b a})$

Mastertheorem

Wir betrachten Gleichungen dieser Form:

$$T(n) = a \cdot T(\frac{n}{b}) + f(n)$$

Drei Fälle werden unterschieden:

1. $f(n) \in O(n^{log_b a - \varepsilon})$ für ein $\varepsilon > 0$
   $\Rightarrow T(n) \in \Theta(n^{log_b a})$
2. $f(n) \in \Theta(n^{log_b a})$
   $\Rightarrow T(n) \in \Theta(n^{log_b a} logn)$

Mastertheorem

Wir betrachten Gleichungen dieser Form:

$$T(n) = a \cdot T(\tfrac{n}{b}) + f(n)$$

Drei Fälle werden unterschieden:

1. $f(n) \in O(n^{log_b a - \varepsilon})$ für ein $\varepsilon > 0$
   $\Rightarrow T(n) \in \Theta(n^{log_b a})$
2. $f(n) \in \Theta(n^{log_b a})$
   $\Rightarrow T(n) \in \Theta(n^{log_b a} log n)$
3. $f(n) \in \Omega(n^{log_b a + \varepsilon})$ für ein $\varepsilon > 0$ und
   $\exists d, 0 < d < 1 : \exists n_0 : \forall n > n_0 : a \cdot f(\tfrac{n}{b}) \leq d \cdot f(n)$
   $\Rightarrow T(n) \in \Theta(f(n))$

Beispiel 1

$$T(n) = 8T\left(\frac{n}{2}\right) + 42n^2$$

Beispiel 1

$$T(n) = 8T(\tfrac{n}{2}) + 42n^2$$

$a = 8, b = 2, f(n) \in \Theta(n^2)$

Beispiel 1

$$T(n) = 8T(\frac{n}{2}) + 42n^2$$

$a = 8, b = 2, f(n) \in \Theta(n^2)$
$log_a b = log_2 8 = 3$

Beispiel 1

$$T(n) = 8T(\frac{n}{2}) + 42n^2$$

$a = 8, b = 2, f(n) \in \Theta(n^2)$
$log_a b = log_2 8 = 3$
$f(n) \in O(n^{log_b a - \varepsilon}) = O(n^{3-\varepsilon})$ mit $\varepsilon = 1$ (1. Fall MT)

Beispiel 1

$$T(n) = 8T(\tfrac{n}{2}) + 42n^2$$

$a = 8, b = 2, f(n) \in \Theta(n^2)$
$log_a b = log_2 8 = 3$
$f(n) \in O(n^{log_b a - \varepsilon}) = O(n^{3-\varepsilon})$ mit $\varepsilon = 1$ (1. Fall MT)
$\Rightarrow T(n) \in \Theta(n^{log_b a}) = \Theta(n^3)$

Beispiel 2

$$T(n) = 2T(\tfrac{n}{2}) + 10n$$

Beispiel 2

$$T(n) = 2T(\tfrac{n}{2}) + 10n$$

$a = 2, b = 2, f(n) \in \Theta(n)$

Beispiel 2

$$T(n) = 2T(\frac{n}{2}) + 10n$$

$a = 2, b = 2, f(n) \in \Theta(n)$
$log_b a = log_2 2 = 1$

Beispiel 2

$T(n) = 2T(\frac{n}{2}) + 10n$

$a = 2, b = 2, f(n) \in \Theta(n)$
$log_b a = log_2 2 = 1$
$f(n) \in \Theta(n^{log_b a}) = \Theta(n)$ (2.Fall MT)

Beispiel 2

$T(n) = 2T(\frac{n}{2}) + 10n$

$a = 2, b = 2, f(n) \in \Theta(n)$
$log_b a = log_2 2 = 1$
$f(n) \in \Theta(n^{log_b a}) = \Theta(n)$ (2.Fall MT)
$\Rightarrow T(n) \in \Theta(n^{log_b a} logn) = \Theta(n \cdot logn)$

Beispiel 3

$$T(n) = 2T(\tfrac{n}{2}) + n^2$$

Beispiel 3

$$T(n) = 2T(\tfrac{n}{2}) + n^2$$

$$a = 2, b = 2, f(n) \in \Theta(n^2)$$

Beispiel 3

$$T(n) = 2T(\frac{n}{2}) + n^2$$

$a = 2, b = 2, f(n) \in \Theta(n^2)$
$log_b a = log_2 2 = 1$

Beispiel 3

$$T(n) = 2T(\frac{n}{2}) + n^2$$

$a = 2, b = 2, f(n) \in \Theta(n^2)$
$log_b a = log_2 2 = 1$
$n^2 \in \Omega(n^{log_b a + \varepsilon}) = \Omega(n^2)$ mit $\varepsilon = 1$

Beispiel 3

$$T(n) = 2T(\frac{n}{2}) + n^2$$

$a = 2, b = 2, f(n) \in \Theta(n^2)$
$log_b a = log_2 2 = 1$
$n^2 \in \Omega(n^{log_b a + \varepsilon}) = \Omega(n^2)$ mit $\varepsilon = 1$
$2(\frac{n}{2})^2 \leq dn^2 \Leftrightarrow \frac{1}{2}n^2 \leq dn^2$

Beispiel 3

$T(n) = 2T(\frac{n}{2}) + n^2$

$a = 2, b = 2, f(n) \in \Theta(n^2)$
$log_b a = log_2 2 = 1$
$n^2 \in \Omega(n^{log_b a + \varepsilon}) = \Omega(n^2)$ mit $\varepsilon = 1$
$2(\frac{n}{2})^2 \leq dn^2 \Leftrightarrow \frac{1}{2}n^2 \leq dn^2$
Mit $d = \frac{1}{2} : \forall n > 1 : \frac{1}{2}n^2 \leq \frac{1}{2}n^2$

Beispiel 3

$$T(n) = 2T(\tfrac{n}{2}) + n^2$$

$a = 2, b = 2, f(n) \in \Theta(n^2)$
$log_b a = log_2 2 = 1$
$n^2 \in \Omega(n^{log_b a + \varepsilon}) = \Omega(n^2)$ mit $\varepsilon = 1$
$2(\frac{n}{2})^2 \leq dn^2 \Leftrightarrow \frac{1}{2}n^2 \leq dn^2$
Mit $d = \frac{1}{2} : \forall n > 1 : \frac{1}{2}n^2 \leq \frac{1}{2}n^2$
$\Rightarrow T(n) \in \Theta(f(n)) = \Theta(n^2)$

Überblick

1. **O-Notation**
2. **Endliche Automaten**
   - Mealy-Automaten
   - f^* und f^{**}
   - g^* und g^{**}
   - Automaten entwerfen
   - Moore-Automaten

Mealy-Automaten

Definition

Ein **Mealy-Automat** ist ein Tupel (Z, X, Y, f, g, z_0) mit

- Z Endliche Zustandsmenge
- X Eingabealphabet
- Y Ausgabealphabet
- f Zustandsübergangsfunktion ($f : Z \times X \to Z$)
- g Ausgabefunktion ($g : Z \times X \to Y^*$)
- z_0 Startzustand ($z_0 \in Z$)

f^* und f^{**}

Definition $f^* : Z \times X^* \to Z$

$$f^*(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^*(z, wx) = f(f^*(z, w), x)$$

In Worten:

f^* und f^{**}

Definition $f^* : Z \times X^* \to Z$

$$f^*(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^*(z, wx) = f(f^*(z, w), x)$$

In Worten: $f^*(z, w)$ ist der Zustand nach Abarbeitung von w beginnend bei Zustand z.

f^* und f^{**}

Definition $f^* : Z \times X^* \to Z$

$$f^*(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^*(z, wx) = f(f^*(z, w), x)$$

In Worten: $f^*(z, w)$ ist der Zustand nach Abarbeitung von w beginnend bei Zustand z.

Definition $f^{**} : Z \times X^* \to Z^*$

$$f^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

In Worten:

f^* und f^{**}

Definition $f^* : Z \times X^* \to Z$

$$f^*(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^*(z, wx) = f(f^*(z, w), x)$$

In Worten: $f^*(z, w)$ ist der Zustand nach Abarbeitung von w beginnend bei Zustand z.

Definition $f^{**} : Z \times X^* \to Z^*$

$$f^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

In Worten: $f^{**}(z, w)$ ist das Wort, das entsteht wenn man alle Zustände bei der Abarbeitung von w hintereinander schreibt.

f^{**} ohne f^*

Definition $f^{**} : Z \times X^* \to Z^*$

$$f^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

In Worten: $f^{**}(z, w)$ ist das Wort, das entsteht wenn man alle Zustände bei der Abarbeitung von w hintereinander schreibt.

Aufgabe

Finde eine alternative Definition für f^{**}, die ohne f^* auskommt.

f^{**} ohne f^*

Definition $f^{**} : Z \times X^* \to Z^*$

$$f^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

In Worten: $f^{**}(z, w)$ ist das Wort, das entsteht wenn man alle Zustände bei der Abarbeitung von w hintereinander schreibt.

Aufgabe

Finde eine alternative Definition für f^{**}, die ohne f^* auskommt.

Lösung

$f^{**}(z, \varepsilon) = z$

f^{**} ohne f^*

Definition $f^{**} : Z \times X^* \to Z^*$

$$f^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

In Worten: $f^{**}(z, w)$ ist das Wort, das entsteht wenn man alle Zustände bei der Abarbeitung von w hintereinander schreibt.

Aufgabe

Finde eine alternative Definition für f^{**}, die ohne f^* auskommt.

Lösung

$f^{**}(z, \varepsilon) = z$
$\forall w \in X^* : \forall x \in X : f^{**}(z, xw) = z \cdot$

f^{**} ohne f^*

Definition $f^{**} : Z \times X^* \to Z^*$

$$f^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : f^{**}(z, wx) = f^{**}(z, w) \cdot f(f^*(z, w), x)$$

In Worten: $f^{**}(z, w)$ ist das Wort, das entsteht wenn man alle Zustände bei der Abarbeitung von w hintereinander schreibt.

Aufgabe

Finde eine alternative Definition für f^{**}, die ohne f^* auskommt.

Lösung

$f^{**}(z, \varepsilon) = z$
$\forall w \in X^* : \forall x \in X : f^{**}(z, xw) = z \cdot f^{**}(f(z, x), w)$

g^* und g^{**}

Definition $g^* : Z \times X^* \to Y^*$

$$g^*(z, \varepsilon) = \varepsilon$$

$$\forall w \in X^* : \forall x \in X : g^*(z, wx) = g(f^*(z, w), x)$$

In Worten:

g^* und g^{**}

Definition $g^* : Z \times X^* \to Y^*$

$$g^*(z, \varepsilon) = \varepsilon$$
$$\forall w \in X^* : \forall x \in X : g^*(z, wx) = g(f^*(z, w), x)$$

In Worten: $g^*(z, w)$ ist die letzte Ausgabe nach Abarbeitung von w beginnend bei Zustand z.

g^* und g^{**}

Definition $g^* : Z \times X^* \to Y^*$

$$g^*(z, \varepsilon) = \varepsilon$$
$$\forall w \in X^* : \forall x \in X : g^*(z, wx) = g(f^*(z, w), x)$$

In Worten: $g^*(z, w)$ ist die letzte Ausgabe nach Abarbeitung von w beginnend bei Zustand z.

Definition $g^{**} : Z \times X^* \to Y^*$

$$g^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : g^{**}(z, wx) = g^{**}(z, w) \cdot g(f^*(z, w), x)$$

In Worten:

g^* und g^{**}

Definition $g^* : Z \times X^* \to Y^*$

$$g^*(z, \varepsilon) = \varepsilon$$
$$\forall w \in X^* : \forall x \in X : g^*(z, wx) = g(f^*(z, w), x)$$

In Worten: $g^*(z, w)$ ist die letzte Ausgabe nach Abarbeitung von w beginnend bei Zustand z.

Definition $g^{**} : Z \times X^* \to Y^*$

$$g^{**}(z, \varepsilon) = z$$
$$\forall w \in X^* : \forall x \in X : g^{**}(z, wx) = g^{**}(z, w) \cdot g(f^*(z, w), x)$$

In Worten: $f^{**}(z, w)$ ist das Wort, das entsteht wenn man alle Ausgaben bei der Abarbeitung von w hintereinander schreibt.

Automaten entwerfen

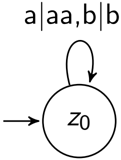
Aufgabe

Gib einen Automaten an, der bei Eingaben aus $\{a, b\}^*$ jedes Zeichen doppelt ausgibt. Also z.B. für *abaa* wird *aabbaaaa* ausgegeben. (Anfangszustand z_0)

Automaten entwerfen

Aufgabe

Gib einen Automaten an, der bei Eingaben aus $\{a, b\}^*$ jedes Zeichen doppelt ausgibt. Also z.B. für *abaa* wird *aabbaaaa* ausgegeben. (Anfangszustand z_0)



Automaten entwerfen

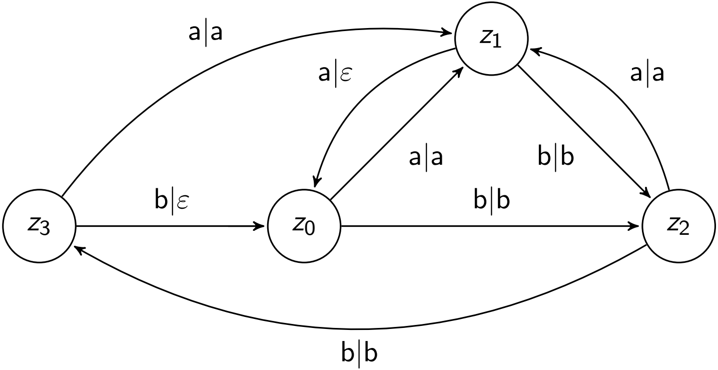
Aufgabe

Gib einen Automaten an, der jedes zweite a und jedes dritte b in Folge aus einer Eingabe aus $\{a, b\}^*$ entfernt. (Anfangszustand z_0)

Automaten entwerfen

Aufgabe

Gib einen Automaten an, der jedes zweite a und jedes dritte b in Folge aus einer Eingabe aus $\{a, b\}^*$ entfernt. (Anfangszustand z_0)



Moore-Automaten

Definition

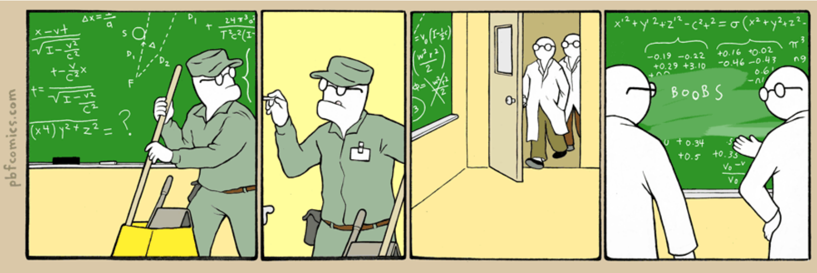
Wie Mealy-Automaten, außer:

- Ausgaben passieren im Zustand, nicht im Zustandsübergang

Mealy- und Moore-Automaten lassen sich einfach ineinander umwandeln.

Überblick

1. **O-Notation**
   - Wiederholung
   - Mastertheorem
2. **Endliche Automaten**
   - Mealy-Automaten
   - f^* und f^{**}
   - g^* und g^{**}
   - Automaten entwerfen
   - Moore-Automaten


pbfcomics.com
BOOBS