

# Suchalgorithmen

## Theorie

Neben dem Sortieren ist das Suchen eine der häufigsten Tätigkeiten, die ein Computer ausführt. Miller und Ranum beschreiben dies in ihrem Buch wie folgt:

*Searching is the algorithmic process of finding a particular item in a collection of items.*

Das Resultat einer Suche kann verschieden ausfallen:

- ▶ Wenn man nur wissen möchte, *ob* das gesuchte Objekt in der Menge vorhanden ist, genügt als Rückgabewert `True` oder `False`.
- ▶ Falls das gesuchte Objekt modifiziert werden soll, möchte man wissen, *wo* das Element innerhalb der Datenstruktur zu finden ist, sofern es überhaupt darin liegt. Möglicherweise tritt ein Wert auch mehrfach auf, so dass mehrere Positionen ermittelt werden müssen.

## Voraussetzung

Die Daten sind in einer Datenstruktur abgelegt, in der – mit Ausnahme des ersten Elements – jedes Element Nachfolger von genau einem anderen Element ist.

Diese Art von Datenorganisation wird in Python durch Listen oder Tupel realisiert. Durch einen Index greift man auf die einzelnen Werte zu.

## Der Algorithmus

Beginnend mit der ersten Position prüft man der Reihe nach jedes Element, bis man entweder gefunden hat, wonach man sucht oder bis man das Arrayende erreicht hat, was bedeutet, dass das Element nicht vorhanden ist.

## Beispiel 2.1

Suche 8 in  $A = [26, 37, 54, 8, 93, 70, 65, 82, 49]$ :

26	37	54	8	93	70	65	82	49	Vergleiche

## Beispiel 2.1

Suche 8 in  $A = [26, 37, 54, 8, 93, 70, 65, 82, 49]$ :

26	37	54	8	93	70	65	82	49	Vergleiche
8									1

## Beispiel 2.1

Suche 8 in  $A = [26, 37, 54, 8, 93, 70, 65, 82, 49]$ :

26	37	54	8	93	70	65	82	49	Vergleiche
8									1
	8								1

## Beispiel 2.1

Suche 8 in  $A = [26, 37, 54, 8, 93, 70, 65, 82, 49]$ :

26	37	54	8	93	70	65	82	49	Vergleiche
8									1
	8								1
		8							1

## Beispiel 2.1

Suche 8 in  $A = [26, 37, 54, 8, 93, 70, 65, 82, 49]$ :

26	37	54	8	93	70	65	82	49	Vergleiche
8									1
	8								1
		8							1
			8						1

## Implementierung in Python

```
1 def linear_search(L, item):  
2     matches = []  
3     for i in range(0, len(L)):  
4         if L[i] == item:  
5             matches.append(i)  
6     return matches
```

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste			
Element $\notin$ Liste			

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste	$O(1)$		
Element $\notin$ Liste			

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste	$O(1)$	$O(n/2) = O(n)$	
Element $\notin$ Liste			

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste	$O(1)$	$O(n/2) = O(n)$	$O(n)$
Element $\notin$ Liste			

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste	$O(1)$	$O(n/2) = O(n)$	$O(n)$
Element $\notin$ Liste	$O(n)$		

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste	$O(1)$	$O(n/2) = O(n)$	$O(n)$
Element $\notin$ Liste	$O(n)$	$O(n)$	

## Laufzeitanalyse der sequentiellen Suche

	Best Case	Average Case	Worst Case
Element $\in$ Liste	$O(1)$	$O(n/2) = O(n)$	$O(n)$
Element $\notin$ Liste	$O(n)$	$O(n)$	$O(n)$

## Voraussetzungen

Die Werte sind ...

- ▶ in einer sequentiellen Datenstruktur abgelegt,
- ▶ in aufsteigender Reihenfolge sortiert.

## Der Algorithmus (binäre Suche)

1. Setze  $a = 0$  und  $b = n - 1$ .
2. Wiederhole, so lange wie  $a \leq b$ :
  - 2.1 Bestimme  $m = \lfloor (a + b) / 2 \rfloor$
  - 2.2 Falls  $e < A[m]$ :  
Setze  $b = m - 1$
  - 2.3 Falls  $e > A[m]$ :  
Setze  $a = m + 1$
  - 2.4 Sonst: # ... gilt  $e = A[m]$ :  
Gib  $m$  als Wert zurück
3. Gib  $-1$  oder `False` zurück

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54					

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$
					54			

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$
					54		1	

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$
					54		1	$a = 4, b = 4, m = 4$

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$
					54		1	$a = 4, b = 4, m = 4$
				54				

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$
					54		1	$a = 4, b = 4, m = 4$
				54			1	

## Beispiel 3.1

Suche 54 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			54				1	$a = 4, b = 6, m = 5$
					54		1	$a = 4, b = 4, m = 4$
				54			1	gefunden

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29					

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$
	29							

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$
	29						1	

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$
	29						1	$a = 2, b = 2, m = 2$

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$
	29						1	$a = 2, b = 2, m = 2$
		29						

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$
	29						1	$a = 2, b = 2, m = 2$
		29					1	

## Beispiel 3.2

Suche 29 in  $A = [8, 26, 37, 49, 54, 65, 70]$ :

0	1	2	3	4	5	6		
8	26	37	49	54	65	70	Vergl.	$a = 0, b = 6, m = 3$
			29				1	$a = 0, b = 2, m = 1$
	29						1	$a = 2, b = 2, m = 2$
		29					1	$a = 2, b = 1$
								nicht gefunden

## Iterative Implementierung in Python

```
1 def search_binary(L, item):
2     lower = 0
3     upper = len(L)-1
4     while (lower <= upper):
5         mid = (lower + upper)//2
6         if item < L[mid]:
7             upper = mid - 1
8         elif item > L[mid]:
9             lower = mid + 1
10        else:
11            return mid
12    return -1 # (semantisch) sinnloser Index
```

## Laufzeitanalyse der binären Suche

Bei jedem Schritt wird die Menge der zu durchsuchenden Elemente etwa halbiert. Im schlimmsten Fall müssen wir das Verfahren so lange durchführen, bis wir eine Liste mit nur noch einem Element haben, die das gesuchte Objekt enthält oder nicht.

Anzahl Schritte	ungefähre Anzahl Elemente
1	$n/2$
2	
...	
$k$	

## Laufzeitanalyse der binären Suche

Bei jedem Schritt wird die Menge der zu durchsuchenden Elemente etwa halbiert. Im schlimmsten Fall müssen wir das Verfahren so lange durchführen, bis wir eine Liste mit nur noch einem Element haben, die das gesuchte Objekt enthält oder nicht.

Anzahl Schritte	ungefähre Anzahl Elemente
1	$n/2$
2	$n/2^2$
...	
$k$	

## Laufzeitanalyse der binären Suche

Bei jedem Schritt wird die Menge der zu durchsuchenden Elemente etwa halbiert. Im schlimmsten Fall müssen wir das Verfahren so lange durchführen, bis wir eine Liste mit nur noch einem Element haben, die das gesuchte Objekt enthält oder nicht.

Anzahl Schritte	ungefähre Anzahl Elemente
1	$n/2$
2	$n/2^2$
...	...
$k$	

## Laufzeitanalyse der binären Suche

Bei jedem Schritt wird die Menge der zu durchsuchenden Elemente etwa halbiert. Im schlimmsten Fall müssen wir das Verfahren so lange durchführen, bis wir eine Liste mit nur noch einem Element haben, die das gesuchte Objekt enthält oder nicht.

Anzahl Schritte	ungefähre Anzahl Elemente
1	$n/2$
2	$n/2^2$
...	...
$k$	$n/2^k$

## Laufzeitanalyse der binären Suche

Bei jedem Schritt wird die Menge der zu durchsuchenden Elemente etwa halbiert. Im schlimmsten Fall müssen wir das Verfahren so lange durchführen, bis wir eine Liste mit nur noch einem Element haben, die das gesuchte Objekt enthält oder nicht.

Anzahl Schritte	ungefähre Anzahl Elemente
1	$n/2$
2	$n/2^2$
...	...
$k$	$n/2^k$

$$n/2^k = 1 \Rightarrow n = 2^k \Rightarrow k = \log_2(n) \Rightarrow T(n) \in O(\log n)$$

## Bemerkung

Damit die binäre Suche angewendet werden kann, müssen die zu durchsuchenden Daten in geordneter Form vorliegen.

Ist dies nicht der Fall, müssen sie zuvor mit einem Sortierverfahren in die richtige Reihenfolge gebracht werden. Die Kosten dafür betragen beim vergleichbasiertem Sortieren mindestens  $O(n \log n)$ .

Da die Laufzeitkomplexität fürs Sortieren bereits grösser als die der sequentiellen Suche, lohnt es sich nicht, die Daten extra zu sortieren, nur um die schnellere binäre Suche anwenden zu können.

Eine weitere zentrale Suchaufgabe besteht darin, ein Textmuster (**pattern**)  $p$  in einer Zeichenkette (**string**)  $t$  zu finden.

Dabei sollen hier Algorithmen betrachtet werden, die nach **exakten** Übereinstimmungen (**matches**) suchen.

Zeichenketten und Muster werden als Listen repräsentiert, deren Elemente die einzelnen Zeichen sind.

# Anwendungen

# Anwendungen

- ▶ Textverarbeitungsprogramme

## Anwendungen

- ▶ Textverarbeitungsprogramme
- ▶ Untersuchung von DNA- und Proteinsequenzen in der Bioinformatik

## Anwendungen

- ▶ Textverarbeitungsprogramme
- ▶ Untersuchung von DNA- und Proteinsequenzen in der Bioinformatik
- ▶ Erkennung von Plagiaten

## Anwendungen

- ▶ Textverarbeitungsprogramme
- ▶ Untersuchung von DNA- und Proteinsequenzen in der Bioinformatik
- ▶ Erkennung von Plagiaten
- ▶ Virens Scanner

## Anwendungen

- ▶ Textverarbeitungsprogramme
- ▶ Untersuchung von DNA- und Proteinsequenzen in der Bioinformatik
- ▶ Erkennung von Plagiaten
- ▶ Virens Scanner

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A												

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B											

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A										

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B									

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B								

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							1

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							1
						A						

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							1
						A	B					

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							1
						A	B	B				

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							1
						A	B	B	A			

## Beispiel 4.1

Suche das Textmuster ABBA in der Zeichenkette ABABBCABBACB

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B										3
	A											1
		A	B	B	A							4
			A									1
				A								1
					A							1
						A	B	B	A			4

## Analyse (Worst Case)

Text: aaaaaaa ( $n = 7$  Zeichen)

Pattern: aab ( $m = 3$  Zeichen,  $m \leq n$ )

a	a	a	a	a	a	a	Vergleiche
a	a	b					3
	a	a	b				3
		a	a	b			3
			a	a	b		3
				a	a	b	3
							$(7-3+1)*3=15$

Allgemein:

$$O((n - m + 1)m) = O(mn - m^2 + m) = O(mn)$$

## Implementierung in Python

```
1 def matcher_naive(pat, text):
2     matches = []
3     n = len(text)
4     m = len(pat)
5     for i in range(0, n-m+1):
6         j = 0
7         while j < m and text[i+j] == pat[j]:
8             j += 1
9         if j==m:
10            matches.append(i)
11    return matches
```

## Schnelle Verschiebungen

Stimmt das Muster an irgend einer Stelle nicht mit dem entsprechenden Teilstring des Textes überein, soll es so weit wie möglich nach rechts verschoben werden, ohne einen Treffer zu verpassen.

```

B B A A B C B B B A A B B A B A A B B B A
A B B A
→ → → A B B A
      → → → → A B B A
                → → → A B B A
                    → A B B A
  
```

Bei Nichtübereinstimmung (rot), kann ich das Muster so weit nach rechts verschieben, bis das letzte Zeichen im Text (blau) mit dem nächsten Zeichen im Muster übereinstimmt. Kommt das Zeichen im Muster nicht vor, kann man sogar um die Länge des Musters verschieben.

## Beispiel 4.2

Suche das Muster ABBA (Pattern  $p$ ) in der Zeichenkette ABABBCABBACB (Text  $t$ ).

## Beispiel 4.2

Suche das Muster ABBA (Pattern  $p$ ) in der Zeichenkette ABABBCABBACB (Text  $t$ ).

Alphabet:  $\Sigma = \{A, B, C\}$  (jedes Symbol in  $t \cup p$ )

## Beispiel 4.2

Suche das Muster ABBA (Pattern  $p$ ) in der Zeichenkette ABABBCABBACB (Text  $t$ ).

Alphabet:  $\Sigma = \{A, B, C\}$  (jedes Symbol in  $t \cup p$ )

**Bad Character Table (BCT):** Um wie viele Positionen darf man das Muster nach rechts verschieben, wenn über seinem rechten Ende im Text das Symbol  $\sigma \in \Sigma$  steht und man keinen Treffer verpassen möchte?

?	?	?	<b>A</b>		
<b>A</b>	B	B	A	$\Rightarrow$	3 Zeichen
<hr/>					
?	?	?	<b>B</b>		
A	B	<b>B</b>	A	$\Rightarrow$	1 Zeichen
<hr/>					
?	?	?	<b>C</b>		
A	B	B	A	$\Rightarrow$	4 Zeichen

## Implementierung der BCT in Python

```
1 def bad_character_table(pat, alph):
2     m = len(pat)
3     D = dict()
4     for i in range(0, len(alph)):
5         D[alph[i]] = m
6     for i in range(0, m-1):
7         D[pat[i]] = m-i-1
8     return D
```

Zeilen 4–5: Jedem Symbol wird provisorisch die Länge des Musters  $m = |p|$  zugeordnet.

Zeile 6–7: Jedem Zeichen im Muster (ausser dem Letzten) wird sein kürzester Abstand vom rechten Ende zugewiesen.

Aufwand:

## Implementierung der BCT in Python

```
1 def bad_character_table(pat, alph):
2     m = len(pat)
3     D = dict()
4     for i in range(0, len(alph)):
5         D[alph[i]] = m
6     for i in range(0, m-1):
7         D[pat[i]] = m-i-1
8     return D
```

Zeilen 4–5: Jedem Symbol wird provisorisch die Länge des Musters  $m = |p|$  zugeordnet.

Zeile 6–7: Jedem Zeichen im Muster (ausser dem Letzten) wird sein kürzester Abstand vom rechten Ende zugewiesen.

Aufwand:  $O(|\Sigma| + m)$

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1
	A	B	B	A								

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1
	A	B	B	A								1

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1
	A	B	B	A								1
		A	B	B	A							

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1
	A	B	B	A								1
		A	B	B	A							1

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1
	A	B	B	A								1
		A	B	B	A							1
						A	B	B	A			

## Beispiel 4.2 (Fortsetzung)

Bad Character Table:

A	B	C
3	1	4

A	B	A	B	B	C	A	B	B	A	C	B	Vergl.
A	B	B	A									1
	A	B	B	A								1
		A	B	B	A							1
						A	B	B	A			4

## Implementierung in Python

```
1 def matcher_bmh(pat, text, alph):
2     matches = []
3     bct = bad_character_table(pat, alph)
4     n = len(text)
5     m = len(pat)
6     i = 0 # Position im Text
7     while (i < n-m+1):
8         j = m-1 # letzte Position im Muster
9         while (j>-1 and pat[j] == text[i+j]):
10             j = j-1
11         if j == -1: # alle Zeichen matchen
12             matches.append(i)
13             i = i + bct[text[i+m-1]] # shift aufgrund BCT
14     return matches
```

## Worst Case-Analyse

Text: aaaaaa ( $n = 6$  Zeichen)

Pattern: baa ( $m = 3$  Zeichen,  $m \leq n$ )

a	a	a	a	a	a	Vergleiche
b	a	a				3
	b	a	a			3
		b	a	a		3
			b	a	a	3
<hr/>						$(6-3+1)*3=12$

Allgemein:  $O((n - m + 1)m) = O(mn - m^2 + m) = O(mn)$

Solche Text-Muster-Strukturen sind jedoch eher die Ausnahme!

## Best Case-Analyse

Text: aaaaaaa ( $n = 6$  Zeichen)

Pattern: bbb ( $m = 3$  Zeichen,  $m \leq n$ )

a	a	a	a	a	a	Vergleiche
b	b	b				1
			b	b	b	1
						$(6//3)*1=2$

Allgemein:  $O(n/m)$

Auch solche Text-Muster-Strukturen sind eher die Ausnahme.

## Average Case-Analyse

Ricardo Baeza-Yates und Mireill Régnier haben in ihrem Artikel in der Fachzeitschrift *Theoretical Computer Science* 1992 gezeigt, dass die Komplexität des Boyer-Moore-Horspool-Algorithmus im Mittel  $O(n)$  ist.

## Bemerkung

Der hier vorgestellten Boyer-Moore-Horspool-Algorithmus (BMH) ist eine Vereinfachung des Boyer-Moore-Algorithmus' (BM), der zusätzlich zur Bad Character Table allfällige Übereinstimmungen am Ende des Musters einbezieht, um es beim ersten Mismatch eventuell noch weiter nach rechts zu verschieben.

Wenn man den Fachartikeln im Internet Glauben schenkt, so ist für natürliche Sprachen der BMH-Algorithmus dem BM-Algorithmus überlegen. Dies liegt offenbar daran, dass der BM-Algorithmus mehr Aufwand zur Verschiebung des Suchmusters betreibt, was sich in einer grösseren Anzahl von Anweisungen niederschlägt.

Siehe z. B.:

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC61442/>

## Die Idee

Wir werden das zu suchende Muster  $p$  als deterministischen endlichen Automaten darstellen, um in linearer Zeit alle akzeptierenden Zustände (Treffer) im Text  $t$  zu finden.

Der Schlüssel zum Verständnis ist die folgende, trivial anmutende Beobachtung:

*Ein Muster  $p$  ist Teilstring eines Textes  $t$ , wenn  $p$  Präfix eines Suffixes von  $t$  ist.*

*Beispiel:* Text:  $t = \text{GCTATCTATGG}$ , Muster:  $p = \text{TAT}$

- ▶  $p = \text{TAT}$  ist Präfix des Suffixes  $\text{TATCTATGG}$  von  $t$
- ▶  $p = \text{TAT}$  ist Präfix des Suffixes  $\text{TATGG}$  von  $t$

## Der Bauplan für den Automaten

- ▶ Die Anzahl der Zeichen dar, die bereits korrekt erkannt wurden, bilden die Zustände  $q_0, q_1, \dots, q_m$  des Automaten.
- ▶ Für jedes Zeichen  $\sigma$  des Alphabets  $\Sigma$  wird ein Übergang  $\delta(q_i, \sigma) = q_j$  definiert.
- ▶ Der Automat akzeptiert genau dann eine Folge von Eingabezeichen, wenn er den Zustand  $q_m$  erreicht hat.
- ▶ Befindet sich der Automat im Zustand  $q_i$  mit  $0 < i < m$  (d. h. er hat bereits einen Teil des Musters korrekt erkannt) und liegt beim nächsten gelesenen Zeichen keine Übereinstimmung mehr vor, dann geht der Automat nur so weit wie nötig zurück. Dazu müssen wir das Muster mit sich selbst vergleichen.

## Beispiel 4.3

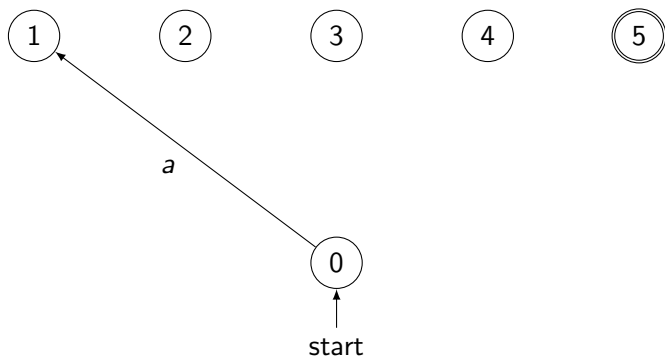
Muster  $p = ababc$



start

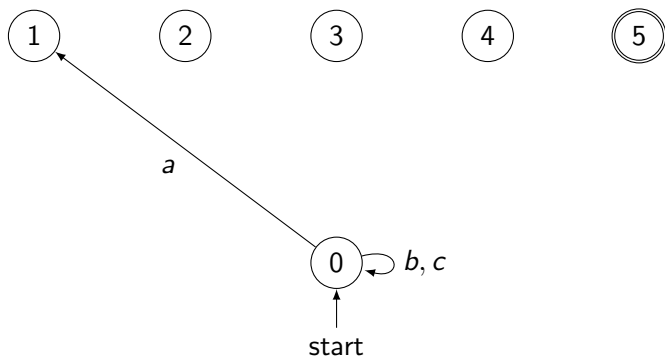
## Beispiel 4.3

Muster  $p = ababc$



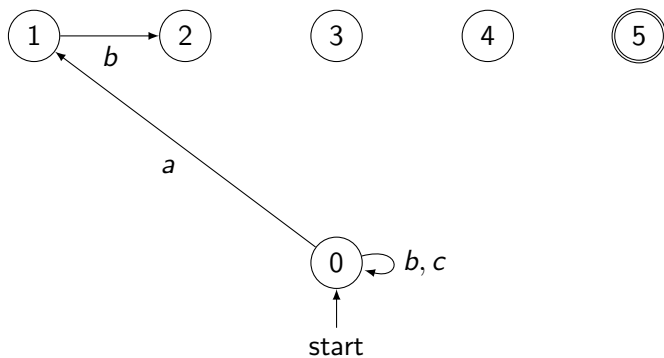
## Beispiel 4.3

Muster  $p = ababc$



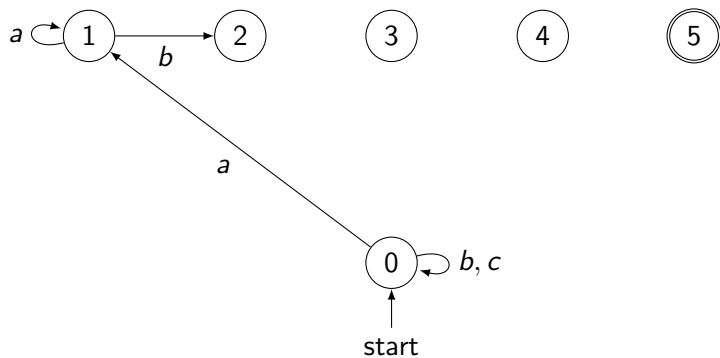
## Beispiel 4.3

Muster  $p = ababc$



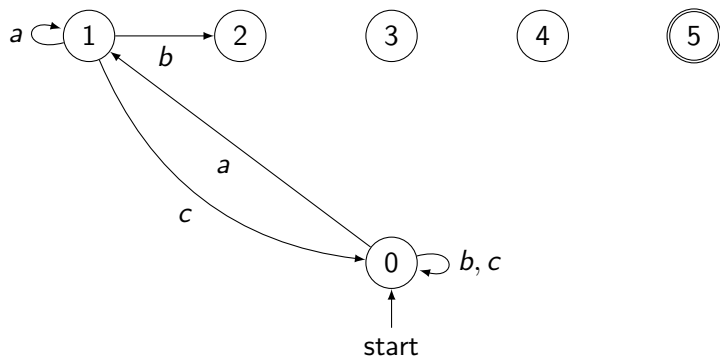
## Beispiel 4.3

Muster  $p = ababc$



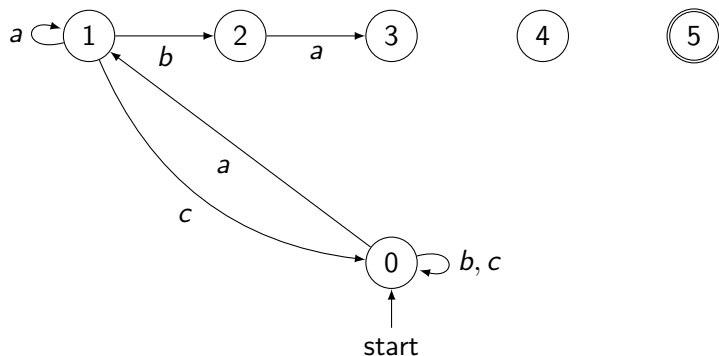
## Beispiel 4.3

Muster  $p = ababc$



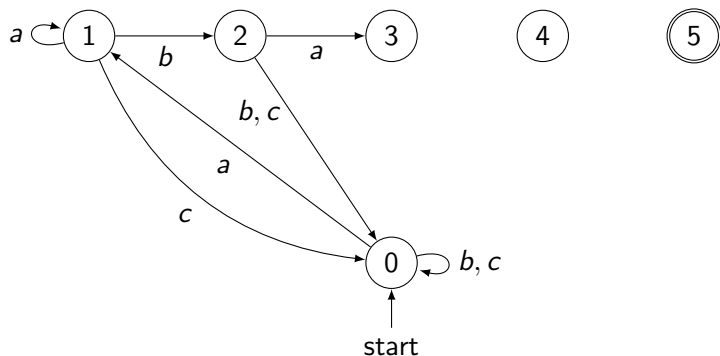
## Beispiel 4.3

Muster  $p = ababc$



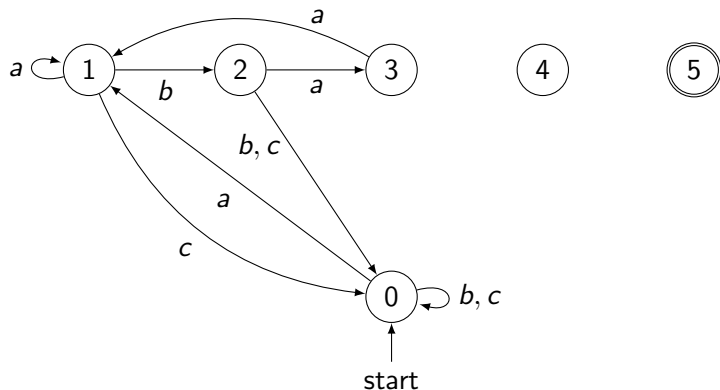
## Beispiel 4.3

Muster  $p = ababc$



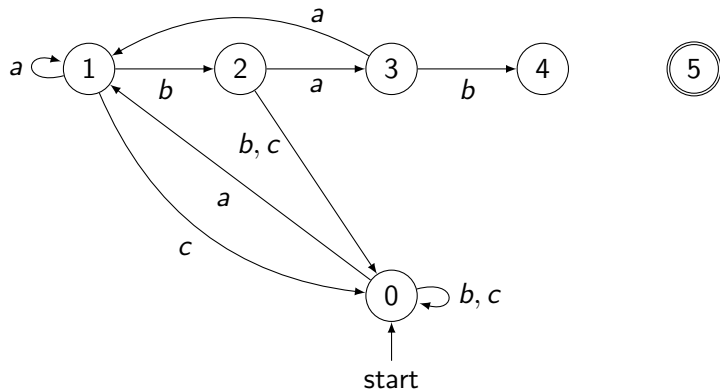
## Beispiel 4.3

Muster  $p = ababc$



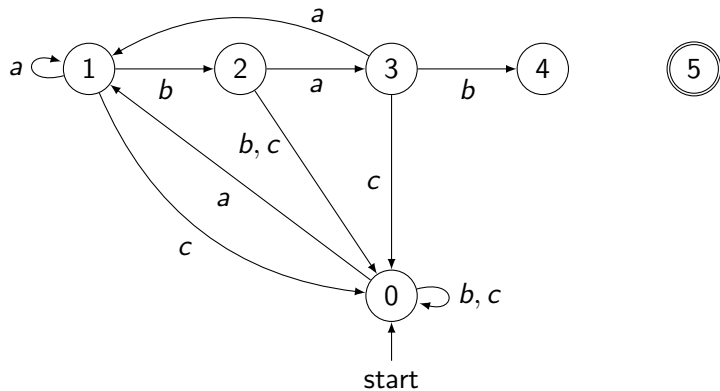
## Beispiel 4.3

Muster  $p = ababc$



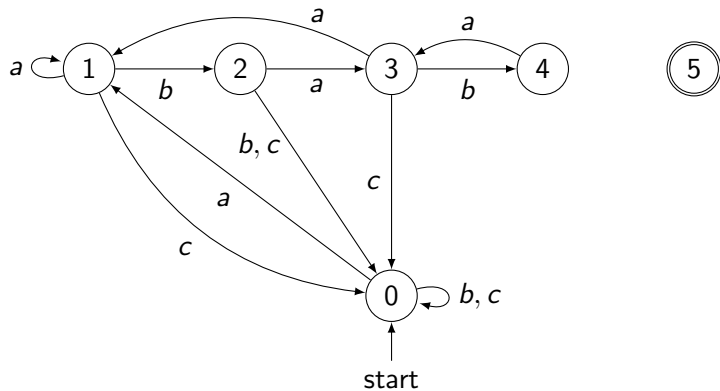
## Beispiel 4.3

Muster  $p = ababc$



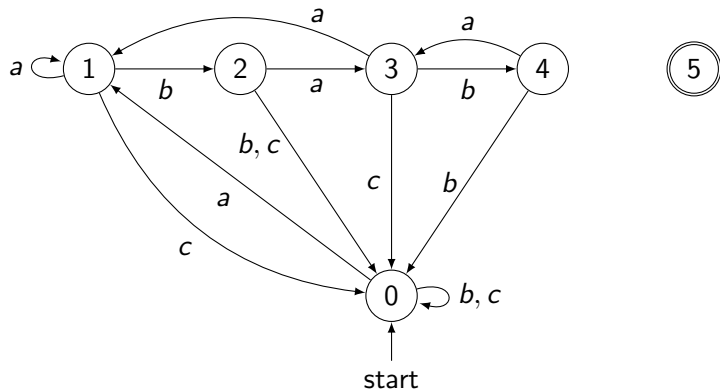
## Beispiel 4.3

Muster  $p = ababc$



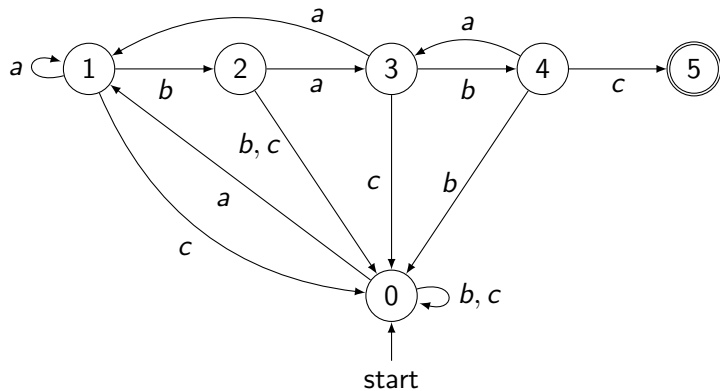
## Beispiel 4.3

Muster  $p = ababc$



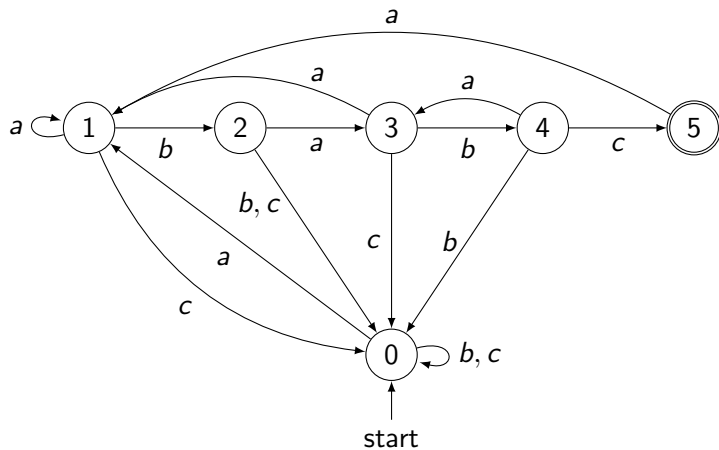
## Beispiel 4.3

Muster  $p = ababc$



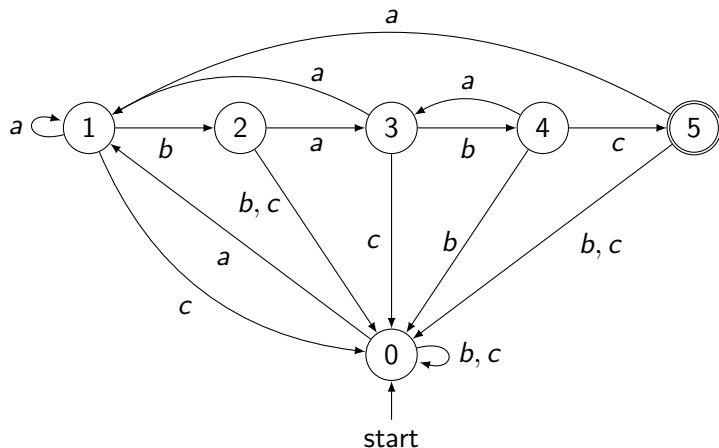
## Beispiel 4.3

Muster  $p = ababc$



## Beispiel 4.3

Muster  $p = ababc$



## Algorithmische Bestimmung des Automaten

Für die systematische Konstruktion des Automaten müssen wir zunächst alle möglichen Präfixe des Musters  $p$  (auch das leere) um jedes Zeichen  $x \in \Sigma$  erweitern.

Dann bestimmen wir das längste Präfix von  $p$ , das gleichzeitig Suffix des um  $x$  erweiterten Präfixes ist.

Die Anzahl der Zeichen in diesem längsten Präfix entspricht dann gerade dem Zustand, den der Automat beim Lesen des Symbols  $x$  erreicht.

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

□  $\varepsilon a$  →

□  $\varepsilon b$  →

□  $\varepsilon c$  →

□  $aa$  →

□  $ab$  →

□  $ac$  →

□  $aba$  →

□  $abb$  →

□  $abc$  →

□  $abaa$  →

□  $abab$  →

□  $abac$  →

□  $ababa$  →

□  $ababb$  →

□  $ababc$  →

□  $ababc a$  →

□  $ababc b$  →

□  $ababc c$  →

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a$	$\square \varepsilon a$	$\rightarrow$	$\square abaa$	$\rightarrow$
	$\square \varepsilon b$	$\rightarrow$	$\square abab$	$\rightarrow$
	$\square \varepsilon c$	$\rightarrow$	$\square abac$	$\rightarrow$
	$\square aa$	$\rightarrow$	$\square ababa$	$\rightarrow$
	$\square ab$	$\rightarrow$	$\square ababb$	$\rightarrow$
	$\square ac$	$\rightarrow$	$\square ababc$	$\rightarrow$
	$\square aba$	$\rightarrow$	$\square ababc a$	$\rightarrow$
	$\square abb$	$\rightarrow$	$\square ababc b$	$\rightarrow$
	$\square abc$	$\rightarrow$	$\square ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a$	$\square \varepsilon a$	$\rightarrow$	<b>1</b>	$\square abaa$	$\rightarrow$
	$\square \varepsilon b$	$\rightarrow$		$\square abab$	$\rightarrow$
	$\square \varepsilon c$	$\rightarrow$		$\square abac$	$\rightarrow$
	$\square aa$	$\rightarrow$		$\square ababa$	$\rightarrow$
	$\square ab$	$\rightarrow$		$\square ababb$	$\rightarrow$
	$\square ac$	$\rightarrow$		$\square ababc$	$\rightarrow$
	$\square aba$	$\rightarrow$		$\square ababc a$	$\rightarrow$
	$\square abb$	$\rightarrow$		$\square ababc b$	$\rightarrow$
	$\square abc$	$\rightarrow$		$\square ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a \rightarrow 1$

$\varepsilon \sqsupset \varepsilon b \rightarrow$

$\sqsupset \varepsilon c \rightarrow$

$\sqsupset aa \rightarrow$

$\sqsupset ab \rightarrow$

$\sqsupset ac \rightarrow$

$\sqsupset aba \rightarrow$

$\sqsupset abb \rightarrow$

$\sqsupset abc \rightarrow$

$\sqsupset aba \rightarrow$

$\sqsupset abab \rightarrow$

$\sqsupset abac \rightarrow$

$\sqsupset ababa \rightarrow$

$\sqsupset ababb \rightarrow$

$\sqsupset ababc \rightarrow$

$\sqsupset ababc a \rightarrow$

$\sqsupset ababc b \rightarrow$

$\sqsupset ababc c \rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a \rightarrow 1$

$\varepsilon \sqsupset \varepsilon b \rightarrow 0$

$\sqsupset \varepsilon c \rightarrow$

$\sqsupset aa \rightarrow$

$\sqsupset ab \rightarrow$

$\sqsupset ac \rightarrow$

$\sqsupset aba \rightarrow$

$\sqsupset abb \rightarrow$

$\sqsupset abc \rightarrow$

$\sqsupset abaa \rightarrow$

$\sqsupset abab \rightarrow$

$\sqsupset abac \rightarrow$

$\sqsupset ababa \rightarrow$

$\sqsupset ababb \rightarrow$

$\sqsupset ababc \rightarrow$

$\sqsupset ababc a \rightarrow$

$\sqsupset ababc b \rightarrow$

$\sqsupset ababc c \rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a \rightarrow 1$

$\varepsilon \sqsupset \varepsilon b \rightarrow 0$

$\varepsilon \sqsupset \varepsilon c \rightarrow$

$\sqsupset aa \rightarrow$

$\sqsupset ab \rightarrow$

$\sqsupset ac \rightarrow$

$\sqsupset aba \rightarrow$

$\sqsupset abb \rightarrow$

$\sqsupset abc \rightarrow$

$\sqsupset abaa \rightarrow$

$\sqsupset abab \rightarrow$

$\sqsupset abac \rightarrow$

$\sqsupset ababa \rightarrow$

$\sqsupset ababb \rightarrow$

$\sqsupset ababc \rightarrow$

$\sqsupset ababc a \rightarrow$

$\sqsupset ababc b \rightarrow$

$\sqsupset ababc c \rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a \rightarrow 1$

$\varepsilon \sqsupset \varepsilon b \rightarrow 0$

$\varepsilon \sqsupset \varepsilon c \rightarrow 0$

$\sqsupset aa \rightarrow$

$\sqsupset ab \rightarrow$

$\sqsupset ac \rightarrow$

$\sqsupset aba \rightarrow$

$\sqsupset abb \rightarrow$

$\sqsupset abc \rightarrow$

$\sqsupset abaa \rightarrow$

$\sqsupset abab \rightarrow$

$\sqsupset abac \rightarrow$

$\sqsupset ababa \rightarrow$

$\sqsupset ababb \rightarrow$

$\sqsupset ababc \rightarrow$

$\sqsupset ababc a \rightarrow$

$\sqsupset ababc b \rightarrow$

$\sqsupset ababc c \rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a \rightarrow 1$

$\varepsilon \sqsupset \varepsilon b \rightarrow 0$

$\varepsilon \sqsupset \varepsilon c \rightarrow 0$

$a \sqsupset aa \rightarrow$

$\sqsupset ab \rightarrow$

$\sqsupset ac \rightarrow$

$\sqsupset aba \rightarrow$

$\sqsupset abb \rightarrow$

$\sqsupset abc \rightarrow$

$\sqsupset abaa \rightarrow$

$\sqsupset abab \rightarrow$

$\sqsupset abac \rightarrow$

$\sqsupset ababa \rightarrow$

$\sqsupset ababb \rightarrow$

$\sqsupset ababc \rightarrow$

$\sqsupset ababc a \rightarrow$

$\sqsupset ababc b \rightarrow$

$\sqsupset ababc c \rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow$	$1$	$\sqcap$	$abaa$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow$	$0$	$\sqcap$	$abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow$	$0$	$\sqcap$	$abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow$	$1$	$\sqcap$	$ababa$	$\rightarrow$
$\sqcap ab$	$\rightarrow$		$\sqcap$	$ababb$	$\rightarrow$
$\sqcap ac$	$\rightarrow$		$\sqcap$	$ababc$	$\rightarrow$
$\sqcap aba$	$\rightarrow$		$\sqcap$	$ababc a$	$\rightarrow$
$\sqcap abb$	$\rightarrow$		$\sqcap$	$ababc b$	$\rightarrow$
$\sqcap abc$	$\rightarrow$		$\sqcap$	$ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a$	$\rightarrow 1$	$\sqsupset aba$	$\rightarrow$
$\varepsilon \sqsupset \varepsilon b$	$\rightarrow 0$	$\sqsupset abab$	$\rightarrow$
$\varepsilon \sqsupset \varepsilon c$	$\rightarrow 0$	$\sqsupset abac$	$\rightarrow$
$a \sqsupset aa$	$\rightarrow 1$	$\sqsupset ababa$	$\rightarrow$
$ab \sqsupset ab$	$\rightarrow$	$\sqsupset ababb$	$\rightarrow$
$\sqsupset ac$	$\rightarrow$	$\sqsupset ababc$	$\rightarrow$
$\sqsupset aba$	$\rightarrow$	$\sqsupset ababca$	$\rightarrow$
$\sqsupset abb$	$\rightarrow$	$\sqsupset ababc b$	$\rightarrow$
$\sqsupset abc$	$\rightarrow$	$\sqsupset ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\sqcap ac$	$\rightarrow$	$\sqcap ababc$	$\rightarrow$
$\sqcap aba$	$\rightarrow$	$\sqcap ababc a$	$\rightarrow$
$\sqcap abb$	$\rightarrow$	$\sqcap ababc b$	$\rightarrow$
$\sqcap abc$	$\rightarrow$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\varepsilon \sqcap ac$	$\rightarrow$	$\sqcap ababc$	$\rightarrow$
$\sqcap aba$	$\rightarrow$	$\sqcap ababc a$	$\rightarrow$
$\sqcap abb$	$\rightarrow$	$\sqcap ababc b$	$\rightarrow$
$\sqcap abc$	$\rightarrow$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\varepsilon \sqcap ac$	$\rightarrow 0$	$\sqcap ababc$	$\rightarrow$
$\sqcap aba$	$\rightarrow$	$\sqcap ababc a$	$\rightarrow$
$\sqcap abb$	$\rightarrow$	$\sqcap ababc b$	$\rightarrow$
$\sqcap abc$	$\rightarrow$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqsupset \varepsilon a \rightarrow 1$$

$$\varepsilon \sqsupset \varepsilon b \rightarrow 0$$

$$\varepsilon \sqsupset \varepsilon c \rightarrow 0$$

$$a \sqsupset aa \rightarrow 1$$

$$ab \sqsupset ab \rightarrow 2$$

$$\varepsilon \sqsupset ac \rightarrow 0$$

$$aba \sqsupset aba \rightarrow$$

$$\sqsupset abb \rightarrow$$

$$\sqsupset abc \rightarrow$$

$$\sqsupset abaa \rightarrow$$

$$\sqsupset abab \rightarrow$$

$$\sqsupset abac \rightarrow$$

$$\sqsupset ababa \rightarrow$$

$$\sqsupset ababb \rightarrow$$

$$\sqsupset ababc \rightarrow$$

$$\sqsupset ababc a \rightarrow$$

$$\sqsupset ababc b \rightarrow$$

$$\sqsupset ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqsupset \varepsilon a$	$\rightarrow 1$	$\sqsupset aba$	$\rightarrow$
$\varepsilon \sqsupset \varepsilon b$	$\rightarrow 0$	$\sqsupset abab$	$\rightarrow$
$\varepsilon \sqsupset \varepsilon c$	$\rightarrow 0$	$\sqsupset abac$	$\rightarrow$
$a \sqsupset aa$	$\rightarrow 1$	$\sqsupset ababa$	$\rightarrow$
$ab \sqsupset ab$	$\rightarrow 2$	$\sqsupset ababb$	$\rightarrow$
$\varepsilon \sqsupset ac$	$\rightarrow 0$	$\sqsupset ababc$	$\rightarrow$
$aba \sqsupset aba$	$\rightarrow 3$	$\sqsupset ababc a$	$\rightarrow$
$\sqsupset abb$	$\rightarrow$	$\sqsupset ababc b$	$\rightarrow$
$\sqsupset abc$	$\rightarrow$	$\sqsupset ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\varepsilon \sqcap ac$	$\rightarrow 0$	$\sqcap ababc$	$\rightarrow$
$aba \sqcap aba$	$\rightarrow 3$	$\sqcap ababca$	$\rightarrow$
$\varepsilon \sqcap abb$	$\rightarrow$	$\sqcap ababc b$	$\rightarrow$
$\sqcap abc$	$\rightarrow$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\varepsilon \sqcap ac$	$\rightarrow 0$	$\sqcap ababc$	$\rightarrow$
$aba \sqcap aba$	$\rightarrow 3$	$\sqcap ababc a$	$\rightarrow$
$\varepsilon \sqcap abb$	$\rightarrow 0$	$\sqcap ababc b$	$\rightarrow$
$\sqcap abc$	$\rightarrow$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\varepsilon \sqcap ac$	$\rightarrow 0$	$\sqcap ababc$	$\rightarrow$
$aba \sqcap aba$	$\rightarrow 3$	$\sqcap ababc a$	$\rightarrow$
$\varepsilon \sqcap abb$	$\rightarrow 0$	$\sqcap ababc b$	$\rightarrow$
$\varepsilon \sqcap abc$	$\rightarrow$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$\sqcap aba$	$\rightarrow$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$\sqcap abab$	$\rightarrow$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\sqcap abac$	$\rightarrow$
$a \sqcap aa$	$\rightarrow 1$	$\sqcap ababa$	$\rightarrow$
$ab \sqcap ab$	$\rightarrow 2$	$\sqcap ababb$	$\rightarrow$
$\varepsilon \sqcap ac$	$\rightarrow 0$	$\sqcap ababc$	$\rightarrow$
$aba \sqcap aba$	$\rightarrow 3$	$\sqcap ababca$	$\rightarrow$
$\varepsilon \sqcap abb$	$\rightarrow 0$	$\sqcap ababc b$	$\rightarrow$
$\varepsilon \sqcap abc$	$\rightarrow 0$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqsupset \varepsilon a \rightarrow 1$$

$$\varepsilon \sqsupset \varepsilon b \rightarrow 0$$

$$\varepsilon \sqsupset \varepsilon c \rightarrow 0$$

$$a \sqsupset aa \rightarrow 1$$

$$ab \sqsupset ab \rightarrow 2$$

$$\varepsilon \sqsupset ac \rightarrow 0$$

$$aba \sqsupset aba \rightarrow 3$$

$$\varepsilon \sqsupset abb \rightarrow 0$$

$$\varepsilon \sqsupset abc \rightarrow 0$$

$$a \sqsupset abaa \rightarrow$$

$$\sqsupset abab \rightarrow$$

$$\sqsupset abac \rightarrow$$

$$\sqsupset ababa \rightarrow$$

$$\sqsupset ababb \rightarrow$$

$$\sqsupset ababc \rightarrow$$

$$\sqsupset ababc a \rightarrow$$

$$\sqsupset ababc b \rightarrow$$

$$\sqsupset ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$\sqcap abab \rightarrow$$

$$\sqcap abac \rightarrow$$

$$\sqcap ababa \rightarrow$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow$$

$$\sqcap abac \rightarrow$$

$$\sqcap ababa \rightarrow$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\sqcap abac \rightarrow$$

$$\sqcap ababa \rightarrow$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow$$

$$\sqcap ababa \rightarrow$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$\sqcap ababa \rightarrow$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$\sqcap ababc \rightarrow$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a$	$\rightarrow 1$	$a \sqcap aba$	$\rightarrow 1$
$\varepsilon \sqcap \varepsilon b$	$\rightarrow 0$	$abab \sqcap abab$	$\rightarrow 4$
$\varepsilon \sqcap \varepsilon c$	$\rightarrow 0$	$\varepsilon \sqcap abac$	$\rightarrow 0$
$a \sqcap aa$	$\rightarrow 1$	$aba \sqcap ababa$	$\rightarrow 3$
$ab \sqcap ab$	$\rightarrow 2$	$\varepsilon \sqcap ababb$	$\rightarrow 0$
$\varepsilon \sqcap ac$	$\rightarrow 0$	$ababc \sqcap ababc$	$\rightarrow$
$aba \sqcap aba$	$\rightarrow 3$	$\sqcap ababc a$	$\rightarrow$
$\varepsilon \sqcap abb$	$\rightarrow 0$	$\sqcap ababc b$	$\rightarrow$
$\varepsilon \sqcap abc$	$\rightarrow 0$	$\sqcap ababc c$	$\rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$ababc \sqcap ababc \rightarrow 5$$

$$\sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$ababc \sqcap ababc \rightarrow 5$$

$$a \sqcap ababc a \rightarrow$$

$$\sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$a \sqcap \varepsilon a \rightarrow 1$	$a \sqcap aba \rightarrow 1$
$\varepsilon \sqcap \varepsilon b \rightarrow 0$	$abab \sqcap abab \rightarrow 4$
$\varepsilon \sqcap \varepsilon c \rightarrow 0$	$\varepsilon \sqcap abac \rightarrow 0$
$a \sqcap aa \rightarrow 1$	$aba \sqcap ababa \rightarrow 3$
$ab \sqcap ab \rightarrow 2$	$\varepsilon \sqcap ababb \rightarrow 0$
$\varepsilon \sqcap ac \rightarrow 0$	$ababc \sqcap ababc \rightarrow 5$
$aba \sqcap aba \rightarrow 3$	$a \sqcap ababc a \rightarrow 1$
$\varepsilon \sqcap abb \rightarrow 0$	$\sqcap ababc b \rightarrow$
$\varepsilon \sqcap abc \rightarrow 0$	$\sqcap ababc c \rightarrow$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$ababc \sqcap ababc \rightarrow 5$$

$$a \sqcap ababc a \rightarrow 1$$

$$\varepsilon \sqcap ababc b \rightarrow$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$ababc \sqcap ababc \rightarrow 5$$

$$a \sqcap ababc a \rightarrow 1$$

$$\varepsilon \sqcap ababc b \rightarrow 0$$

$$\sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$ababc \sqcap ababc \rightarrow 5$$

$$a \sqcap ababc a \rightarrow 1$$

$$\varepsilon \sqcap ababc b \rightarrow 0$$

$$\varepsilon \sqcap ababc c \rightarrow$$

## Beispiel 4.4 (systematische Konstruktion)

Muster:  $p = ababc$

$$a \sqcap \varepsilon a \rightarrow 1$$

$$\varepsilon \sqcap \varepsilon b \rightarrow 0$$

$$\varepsilon \sqcap \varepsilon c \rightarrow 0$$

$$a \sqcap aa \rightarrow 1$$

$$ab \sqcap ab \rightarrow 2$$

$$\varepsilon \sqcap ac \rightarrow 0$$

$$aba \sqcap aba \rightarrow 3$$

$$\varepsilon \sqcap abb \rightarrow 0$$

$$\varepsilon \sqcap abc \rightarrow 0$$

$$a \sqcap abaa \rightarrow 1$$

$$abab \sqcap abab \rightarrow 4$$

$$\varepsilon \sqcap abac \rightarrow 0$$

$$aba \sqcap ababa \rightarrow 3$$

$$\varepsilon \sqcap ababb \rightarrow 0$$

$$ababc \sqcap ababc \rightarrow 5$$

$$a \sqcap ababc a \rightarrow 1$$

$$\varepsilon \sqcap ababc b \rightarrow 0$$

$$\varepsilon \sqcap ababc c \rightarrow 0$$

## Funktionsweise des Automaten

Wenn einmal der DFA aus dem Muster  $p$  erzeugt wurde, ist es eine einfache Aufgabe, nach dem Vorkommen von  $p$  in einem Text  $t$  zu suchen: Man liest den Text  $t$  zeichenweise ein und „füttert“ damit den DFA, der zuvor in den Anfangszustand  $q_0$  gesetzt wurde.

Abhängig vom gelesenen Zeichen führt der DFA einen Zustandswechsel durch oder verbleibt im gleichen Zustand. Gerät der DFA nach dem  $i$ -ten gelesenen Zeichen in den akzeptierenden Zustand, so hat er das Ende des Musters erreicht und „weiss“, dass das Muster an der Position  $i - |p| + 1$  im Text  $t$  vorkommt, wobei  $|p|$  für die Länge des Musters  $p$  steht.

## Beispiel 4.5

Wir suchen im Text  $t = \text{aaababcababcc}$  nach dem Muster  $p = \text{ababc}$ , wobei wir den in Beispiel 4.3 bzw. Beispiel 4.4 konstruierten DFA verwenden.

												1	1	1
Index	-	0	1	2	3	4	5	6	7	8	9	0	1	2
Symbol:	$\varepsilon$	a	a	a	b	a	b	c	a	b	a	b	c	c
State:	0	1	1	1	2	3	4	5	1	2	3	4	5	0
Match:	-	-	-	-	-	-	-	2	-	-	-	-	7	-

## Generierung des DFAs in Python

```
1 def build_dfa(pat, alph):
2     '''Gibt einen DFA aus Muster und Alphabet zurück.'''
3     m = len(pat)
4     delta = [dict() for i in range(0, m+1)]
5
6     for q in range(0, m+1):
7         for x in alph:
8             s1 = pat[:q] + x
9             for k in range(m, -1, -1):
10                s2 = pat[:k]
11                if s1.endswith(s2):
12                    delta[q][x] = k
13                    break
14
15     return delta
```

## Mustersuche mit einem DFA in Python

```
1 def matcher_dfa(pat, text, alph):
2     '''Gibt eine Liste mit den Anfangspositionen
3     der Matches zurück.'''
4     matches = []
5     delta = build_dfa(pat, alph)
6     q = 0
7     m = len(pat)
8     for i, c in enumerate(text):
9         q = delta[q][text[i]]
10        if q == m:
11            matches.append(i-m+1)
12    return matches
```