

Stringmatching

Übungen

Aufgabe 1

Bestimme die Anzahl der Vergleiche, die der „naive“ Algorithmus für das String-Matching zum Auffinden des Musters GGCA im Textstring GGGAAAGGCAT benötigt.

Aufgabe 1

G	G	G	A	A	A	G	G	C	A	T	Vergleiche
G	G	C									3
	G	G	C								3
		G	G								2
			G								1
				G							1
					G						1
						G	G	C	A		4
Total											15

Aufgabe 2

Erstelle die Boyer-Moore-Bad-Character-Tabelle für das Suchmuster SALATTELLER zuerst auf „intuitive“ Art und danach durch den im Python-Code implementierten Algorithmus. Zeichen des Alphabets, die nicht im Suchmuster vorkommen, sind mit einem Stern (*) zu berücksichtigen.

Aufgabe 2

Intuitiv:

S	A	L	T	E	R	*
10	7	2	5	1	11	11

Algorithmisch:

Zählhilfe SALATTELLER (Länge: 11)
0987654321-

S	A	L	T	E	R	*
11	11	11	11	11	11	11
10	9	8	6	4		
	7	3	5	1		
		2				

Aufgabe 3

Bestimme die Anzahl der Vergleiche, die der Boyer-Moore-Horspool-Algorithmus für das String-Matching zum Auffinden des Musters GGCA im Textstring GGGAAAGGCAT benötigt.

Aufgabe 3

pattern=GGCA ($m = 4$)

Bad Character Table:

Character	G	C	A	*
Shift	2	1	4	4

Das Symbol * steht für alle Buchstaben, die nicht im Suchmuster vorkommen.

Shift = Wert[pattern[j]] = $m-j-1$ ($j=0, \dots, m-2$)

G	G	G	A	A	A	G	G	C	A	T	Vergleiche
G	G	C	A								2
				G	G	C	A				1
						G	G	C	A		4
Total											7

Aufgabe 4

Gib den deterministischen endlichen Automaten an (graphisch oder als Tabelle), mit dem in einem beliebigen Text nach dem Muster ABAB gesucht werden kann.

Aufgabe 4

s ist das längste Suffix von $p[0:i]+x$, das Präfix des Suchmusters $p = ABAB$ ist.

q_i	$p[0:i]+x$	s	q_{i+1}
0	A	A	1
0	B	ϵ	0
1	AA	A	1
1	AB	AB	2
2	ABA	ABA	3
2	ABB	ϵ	0
3	ABAA	A	1
3	ABAB	ABAB	4
4	ABABA	ABA	3
4	ABABB	ϵ	0

Der DFA in graphischer Darstellung:

