

Aufgabe 1

Ein *Suchverfahren* ist ein Algorithmus, der in einem Suchraum Objekten mit bestimmten Eigenschaften sucht.

Arten von Suchalgorithmen

- Suche in ungeordneten Listen
- Suche in geordneten Listen
- Suche in Bäumen (wurde bisher nicht explizit behandelt)
- Suche in Graphen (Tiefen- und Breitensuche)
- Suche in Zeichenketten (naiv, Boyer-Moore-Horspool, DFA)

Aufgabe 2

- (a) Suche in einer unsortierten Liste mit n Elementen

$$O(n)$$

- (b) Suche in einer sortierten Liste mit n Elementen

$$O(\log n)$$

- (c) Tiefensuche in einem Graphen mit $|V|$ Knoten und $|E|$ Kanten in Adjazenzlisten-darstellung

$$O(|E| + |V|)$$

- (d) Breitensuch in einem Graphen mit $|V|$ Knoten und $|E|$ Kanten in Adjazenzlisten-darstellung

$$O(|E| + |V|)$$

- (e) Naive Suche nach einer Zeichenkette mit m Zeichen in einer Zeichenkette mit n Zeichen

$$m \cdot n$$

Aufgabe 3

(a) Python-Code:

1	d	e	f		i	s	_	e	l	e	m	e	n	t	_	o	f	(L	,	i	t	e	m)	:		
2			f	o	r				i		i	n		r	a	n	g	e	(l	e	n	(L))	:	
3					i	f			L	[i]	=	=	i	t	e	m	:									
4									r	e	t	u	r	n		T	r	u	e									
5			r	e	t	u	r	n		F	a	l	s	e														
6																												
7																												
8																												

(b) Genau genommen müssen zwei Fälle unterschieden werden.

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

- (c)
- Initialisiere zwischen Zeile 1 und Zeile 2 eine leere Liste $I=[]$.
 - Ersetze `return True` durch `I.append(i)`.
 - Ersetze `return False` durch `return I`.

Aufgabe 5

(a) G G G A A A G G C A T	Vergleiche
G G C A	3
G G C A	3
G G C A	2
G G C A	1
G G C A	1
G G C A	1
G G C A	4
Total	15

(b) $t = \text{aaaaaaa}$, $p = \text{aab}$

```

a a a a a a a
a a b
  a a b
    a a b
      a a b
        a a b

```

Das Worst-Case-Muster der Länge $m = 3$ wird $n - m + 1 = 7 - 3 + 1 = 5$ mal erfolglos mit dem jeweiligen Textabschnitt verglichen.

Somit gilt: $O((n - m + 1)m) = O(nm - m^2 + m)$

Gehen wir davon aus, dass der Text sehr viel länger als das Muster ist, ist, können wir die Summanden m und m^2 gegenüber n vernachlässigen und erhalten $O(nm)$.

Aufgabe 6

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, ..., 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
<hr/>											
Total											7

Aufgabe 7

Suchmuster: SALATTELLER (Länge: 11)

S	A	L	T	E	R	*	
11	11	11	11	11	11	11	Initialisierung
10	11	11	11	11	11	11	$d(S, R) = 11 - 1 = 10$
10	9	11	11	11	11	11	$d(A, R) = 11 - 2 = 9$
10	9	8	11	11	11	11	$d(L, R) = 11 - 3 = 8$
10	7	8	11	11	11	11	$d(A, R) = 11 - 4 = 7$
10	7	8	6	11	11	11	$d(T, R) = 11 - 5 = 6$
10	7	8	5	11	11	11	$d(T, R) = 11 - 6 = 5$
10	7	8	5	4	11	11	$d(E, R) = 11 - 7 = 4$
10	7	3	5	4	11	11	$d(L, R) = 11 - 8 = 3$
10	7	2	5	4	11	11	$d(L, R) = 11 - 9 = 2$
10	7	2	5	1	11	11	$d(E, R) = 11 - 10 = 1$

