

Algorithmen & Datenstrukturen (AD)

## Übung: Suchalgorithmen (A4)

Themen: Suchalgorithmen, einfache Suche, optimierter Suchautomat, Rand eines Wortes, Musterautomat, KMP-Algorithmus, Quicksearch, Optimal-Mismatch.  
Zeitbedarf: ca. 300min.

Hansjörg Diethelm, Version 1.00 (FS 2017)

---

### 1 Einfache Suche (ca. 15')

#### 1.1 Lernziel

- Die einfache Suche verstehen.
- Anhand von Extrembeispielen die Leistungsfähigkeit bzw. die Laufzeitkomplexität abschätzen/einordnen können.

#### 1.2 Grundlagen

Diese Aufgabe basiert auf dem AD-Input "A41\_IP\_Suchalgorithmen". Die Aufgabe beinhaltet keine Programmierung.

#### 1.3 Aufgabe

Die Zeichenkette "aaaaaaaaaaaaaaaaaaaaa...a" mit  $n = 100'000$  soll mit Hilfe einer einfachen Suche nach einem Pattern mit der Länge  $m = 10$  durchsucht werden.

- a) Wie lautet ein konkretes Pattern, so dass die Anzahl Zeichenvergleiche für die Suche minimal wird? Wie viele Zeichenvergleiche sind es in diesem Fall genau?
- b) Wie lautet ein konkretes Pattern, so dass die Anzahl Zeichenvergleiche maximal wird? Wie viele Zeichenvergleiche sind es dann genau?
- c) Welche Laufzeitkomplexität besitzt die "einfache Suche"?

## 2 Optimierter Suchautomat (ca. 90')

### 2.1 Lernziele

- Einen optimierten Suchautomaten für ein gegebenes Pattern herleiten/aufzeichnen.
- Einen optimierten Suchautomaten implementieren.
- Merkmale des optimierten Suchautomaten kennen.

### 2.2 Grundlagen

Diese Aufgabe basiert auf dem AD-Input "A41\_IP\_Suchalgorithmen".

### 2.3 Aufgaben

- a) Welche Vorteile hat das Suchen mit Hilfe eines optimierten Suchautomaten im Vergleich zu einer einfachen Suche?
- b) Zeichnen Sie einen optimierten Suchautomaten für das Suchen nach dem Pattern "ANANAS" auf. Das Alphabet beinhalte folgende Symbole  $A = \{'A', 'N', 'S', \dots\}$ .
- c) Zeichnen Sie einen optimierten Suchautomaten für das Suchen nach dem Pattern "EISBEIN" auf. Das Alphabet beinhalte folgende Symbole  $A = \{'B', 'E', 'I', 'N', 'S', \dots\}$ .
- d) Inwiefern werden sich die Implementierungen für die Suche nach "ANANAS" bzw. "EISBEIN" prinzipiell von jener nach dem binären Pattern "101011" aus dem Input unterscheiden?
- e) Implementieren Sie eine Methode  

```
public static int stateSearch(final String a)
```

welche nach dem Pattern "ANANAS" sucht. Testen Sie Ihre Methode.

### 3 Musterautomat und KMP-Algorithmus (ca. 75')

#### 3.1 Lernziele

- Einen Musterautomaten für ein gegebenes Pattern herleiten/aufzeichnen.
- Arbeitsweise und Merkmale des KMP-Algorithmus kennen.

#### 3.2 Grundlagen

Diese Aufgabe basiert auf dem AD-Input "A41\_IP\_Suchalgorithmen".

#### 3.3 Aufgaben

- Bestimmen Sie die Ränder aller Teilworte von "EISBEIN".
- Zeichnen Sie den Musterautomaten für das Pattern "EISBEIN" auf.
- Lösen Sie obige Aufgaben a) und b) analog für "ANANAS".
- Dem Input können Sie den Musterautomaten zum Pattern "abcaab" entnehmen. Arbeiten Sie den Automaten gedanklich und "auf Papier" schrittweise ab, und zwar für das Durchsuchen folgender Zeichenkette, bis das Pattern gefunden wurde:  
  
a b c a a a b c a c a a b c a a b a a b a b c a b c a a a
- Beim KMP-Algorithmus liefert die Methode `initNext(...)` als Ergebnis ein Array `int[]` zurück. Mit welchen Zahlenwerten wird dieses Array belegt sein, wenn wir nach dem Pattern "EISBEIN" bzw. "ANANAS" suchen?
- Führen Sie den KMP-Algorithmus aus und verifizieren Sie Ihre Antworten von e). Jetzt können Sie natürlich nach beliebigen Patterns suchen!

## 4 Quicksearch (ca. 30')

### 4.1 Lernziele

- Einen Musterautomaten für ein gegebenes Pattern herleiten/aufzeichnen.
- Arbeitsweise und Merkmale des Quicksearch-Algorithmus kennen.

### 4.2 Grundlagen

Diese Aufgabe basiert auf dem AD-Input "A41\_IP\_Suchalgorithmen". Die Aufgabe beinhaltet keine Programmierung.

### 4.3 Aufgaben

- a) Woraus ergibt sich die Länge für das `shift`-Array? In welchem Bereich (min ... max) liegen die Zahlenwerte im `shift`-Array?
- b) Es gilt folgendes Alphabet:  $A = \{a, b, c, d, e, f, g, h, i\}$ . Wir wollen mit Quicksearch nach dem Pattern "gaga" suchen. Mit welchen Werten muss dafür das Array `shift` initialisiert werden?
- c) Geben Sie zwei Extrembeispiele an, d.h. je eine Zeichenkette und ein Pattern, wo Quicksearch
  - extrem schnell durch eine Zeichenkette "springt" bzw.
  - langsam durch eine Zeichen "kriecht".

Schätzen Sie die Anzahl erforderliche Zeichenvergleiche für diese Spezialfälle verallgemeinert ab, und zwar in der Annahme, die Zeichenkette enthalte das Pattern nicht.

## 5 Quicksearch und Optimal-Mismatch (ca. 60')

### 5.1 Lernziele

- Arbeitsweise und Merkmale des Quicksearch versus des Optimal-Mismatch Algorithmus kennen.

### 5.2 Grundlagen

Diese Aufgabe basiert auf dem AD-Input "A41\_IP\_Suchalgorithmen".

### 5.3 Aufgaben

- a) Implementieren und testen Sie den Quicksearch-Algorithmus mit Hilfe adäquater und überschaubarer Testfälle.
- b) Machen Sie ein paar Zeitmessungen beim Durchsuchen langer Zeichenketten bzw. grosser Texte. Vielleicht lesen Sie dafür eine grosse Textdatei ein.
- c) Modifizieren Sie den Code von Quicksearch zum Optimal-Mismatch Algorithmus. Dazu sind nur wenige Zeilen Code erforderlich. Entscheiden Sie sich für eine der vorgestellten Strategien. Jetzt können Sie die Testfälle von a) wieder gebrauchen.
- d) Machen Sie erneut Zeitmessungen, idealerweise so, dass Sie diese mit jenen von b) vergleichen können.

## 6 Übersicht Suchalgorithmen (ca. 30')

### 6.1 Lernziel

- Behandelte Suchalgorithmen reflektieren.
- Merkmale der behandelten Suchalgorithmen kennen.

### 6.2 Grundlagen

Diese Aufgabe basiert auf dem AD-Input "A41\_IP\_Suchalgorithmen". Die Aufgabe beinhaltet keine Programmierung.

### 6.3 Aufgabe

- a) Erstellen Sie gemäss folgender Vorlage eine Übersicht. Ihre Einträge sollten Sie nicht nur "stur" zusammentragen, sondern möglichst auch verstehen!

Eigenschaften:	Zeitkomplexität			Zeichenkette rein sequentiell lesbar (ja/nein)	spezielle Merkmale und Hinweise
	Best Case	Average Case	Worst Case		
<b>Algorithmus:</b>					
Einfache Suche					
Optimierter Suchautomat					
KMP-Algorithmus (Musterautomat)					
Quicksearch					
Optimal-Mismatch					

- b) Optional: Hier gibt's noch Gelegenheit zum Experimentieren und zum Entdecken:

<http://www-igm.univ-mlv.fr/~lecroq/string/>