Prof. Dr. Klaus Volbert

Regensburg, den 09. Dezember 2025

Votieren und abgeben bis zum 15. Dezember 2025, 23:00 Uhr

Übungen zur Vorlesung

Algorithmen und Datenstrukturen

WiSe 2025/26

Blatt 10

Wichtige Hinweise:

- > Falls Sie bei der Bearbeitung einer Aufgabe größere Schwierigkeiten hatten und deswegen die Bearbeitung abgebrochen haben, so versuchen Sie bitte Ihre Schwierigkeiten in Form von Fragen festzuhalten. Bringen Sie Ihre Fragen einfach zur Vorlesung oder zur Übung mit!
- > Musterlösungen werden bei Bedarf in den Übungen besprochen!

Aufgabe 1:
Realisieren Sie die Datenstruktur Skip-Liste in C,C++, C#, Java oder Python. Jeder Schlüssel soll maximal einmal in einer Skip-Liste enthalten sein und folgende Operationen sollen mindestens zur Verfügung stehen:

- `Init()`: Initialisiert eine leere Skip-Liste
- `Deinit()`: Deinitialisiert eine Skip-Liste
- `Print()`: Gibt die aktuelle Skip-Liste aus (z.B. auf der Konsole)
- `Insert(key k)`: Fügt einen Schlüssel k in die Skip-Liste ein
- `Delete(key k)`: Entfernt einen Schlüssel k aus der Skip-Liste
- `Search(key k)`: Gibt `true` zurück, wenn der Schlüssel k in der Skip-Liste vorhanden ist, sonst `false`

Testen Sie Ihre Implementierung anhand verschiedener Zufallszahlenfolgen.

Aufgabe 2:
Suchen Sie das Muster P=„DATEN“ in dem Text T=„ALGORITHMEN UND DATENSTRUKTUREN“ mittels der in der Vorlesung vorgestellten Algorithmen `NaiveSearch` und `BoyerMooreSearch`. Markieren Sie jeweils die durchgeführten Vergleiche.

Aufgabe 3:
Zeigen Sie, dass der Algorithmus `NaiveSearch` im Average Case Laufzeit $\Theta(n)$ hat. Nehmen Sie hierzu an, dass sowohl der Text der Länge n als auch das Muster der Länge m zufällig und gleichverteilt aus einem Alphabet mit $|\Sigma| \geq 2$ Buchstaben gewählt werden. Die Aussage

folgt sofort, wenn Sie begründen, dass die erwartete Anzahl an Buchstaben-Vergleichen durch folgende Formel gegeben ist:

$$(n-m+1)\frac{1-|\Sigma|^{-m}}{1-|\Sigma|^{-1}} \leq 2(n-m+1) = \Theta(n)$$

Aufgabe 4:
Verbessern Sie den Algorithmus `NaiveSearch` unter der Annahme, dass alle Zeichen in dem Muster P verschieden sind, so dass er eine Laufzeit von $O(n)$ garantiert.