

## Arbeitsblatt: Open Hashing

### Aufgabe 1 Clustering

Sei  $n$  die Tabellengrösse und  $k$  die Grösse eines Clusters in der Tabelle. Beantworten Sie folgende Fragen zu linearem Sondieren:

- a) Wie hoch ist die Wahrscheinlichkeit, dass beim nächsten Einfügen eines neuen Elements der Cluster wächst?
  
  
  
  
  
  
  
  
  
  
- b) Wie hoch ist die Wahrscheinlichkeit, dass das Einfügen eines neuen Elements mindestens 3 Sondierungs-Schritte benötigt (d.h. dreimal ist das Feld bereits besetzt)?
  
  
  
  
  
  
  
  
  
  
- c) Nehmen Sie nun an, die Tabelle hat Grösse 23 und drei Cluster der Grösse 5. Wie viele Sondierungs-Schritte erwarten Sie im Schnitt beim Einfügen eines neuen Elements?

### Aufgabe 2 Step

Double Hashing ist besser als Lineares Sondieren, weil die Sondierungsschrittweite für verschiedene Elemente verschieden gewählt wird. Warum genügt es nicht, einfach z.B.  $\text{step} = 7$  für alle Elemente zu wählen? Begründen Sie Ihre Antwort mit einem Beispiel.

**Aufgabe 3 Load Factor**

Sei  $L$  der Load Factor der Hashtabelle. Beantworten Sie folgende Fragen zu Separate Chaining. Nehmen Sie an, dass die Schlüssel zufällig verteilt sind.

- Was ist die erwartete Länge einer Liste in der Tabelle?
- Wie hoch ist der erwartete Aufwand einer erfolgreichen Suche (d.h. das Element ist in der Hashtabelle enthalten)?
- Wie hoch ist der erwartete Aufwand einer erfolglosen Suche?

**Aufgabe 4 Hands-On**

- In eine leere Hashtabelle der Grösse 11, fügen Sie die untenstehenden Schlüssel ein.

31, 45, 15, 99, 11, 5, 47, 4, 14, 28

Benutzen Sie dafür einmal lineares Sondieren mit  $H(x) = x \% 11$  und einmal Double Hashing mit zweiter Hashfunktion  $H2(x) = 1 + x \% 9$ .

lineares Sondieren:

0	1	2	3	4	5	6	7	8	9	10

double Hashing:

0	1	2	3	4	5	6	7	8	9	10

- Löschen Sie nun aus beiden Tabellen die Schlüssel 99 und 11 und fügen Sie danach den Schlüssel 23 ein.