

# Prüfung Algorithmen und Datenstrukturen

## Allgemeine Hinweise:

- Laden Sie sich zusätzlich zu dieser Angabe die Textdatei mit dem Titel "loesungsbogen.txt" als Vorlage für Ihre Abgabe herunter, in die Sie Ihre Lösungen zu den Aufgaben der Prüfung eintragen.
- Nutzen Sie bitte ausschließlich die .txt-Datei ("loesungsbogen.txt"), um ihre Lösung abzugeben.
- Die Art und Weise, eine Aufgabe zu lösen, wird ihnen detailliert auf der Angabe beschrieben. Lesen Sie diese daher bitte genau.
- Sie bekommen ab Beginn der Prüfung 180 Minuten Zeit, diese herunterzuladen, zu bearbeiten und Ihre Lösung wieder hochzuladen (alles via Uni2work). Somit haben Sie 90 Minuten extra Zeit, um auf eventuelle technische Probleme reagieren zu können.
- Speichern Sie vor Bearbeitung der Klausur die Lösungsdatei mit dem Titel "loesungsbogen\_[Matrikelnummer].txt" an einem Ort ab, an dem Sie sie jederzeit wiederfinden. Ersetzen Sie dabei [Matrikelnummer] mit ihrer Matrikelnummer.
- Vergessen Sie nicht, die Lösungsdatei möglichst häufig Zwischenspeichern. So reduzieren Sie Stress durch technische Probleme am Ende der Abgabezeit.
- Notfall-Hotline: 089 / 2180 9306

- Ich habe die Lösung eigenständig und ohne die Hilfe Dritter angefertigt.
- Ich bin der rechtmäßige Nutzer dieses Uni2Work-Accounts und gebe die Lösung nicht für jemand Drittes ab.
- Ich bin zum Zeitpunkt der Prüfung immatrikuliert und kann dies auch durch ein entsprechendes Dokument jederzeit während des Prüfungsprozesses bestätigen.
- Ich veröffentliche keinerlei Prüfungsdokumente wie Aufgabenstellung, Korrektur, etc. im öffentlichen Raum (Online, Aushang, Bekanntenkreis,...).
- Ich Sorge dafür, dass ich regelmäßige Updates meiner Lösungen in Uni2Work mache, sodass es zu verminderten technischen Problemen am Ende kommt. Die letzte innerhalb der Abgabefrist hochgeladene Version wird korrigiert. Achtung: Uni2Work logged Sie nach einer gewissen Zeit der Inaktivität automatisch wieder aus!

Die Prüfung besteht aus 7 Aufgaben. Die Punktezahl ist bei jeder Aufgabe angegeben.

Aufgabe	mögliche Punkte	erreichte Punkte
1.Allgemeine Fragen	18	
2.Gemischte Aufgaben	15	
3.Suchbäume	9	
4.Graphalgorithmen	16	
5.Algorithmenanalyse	9	
6.Sortieralgorithmen	5	
7.Hashing	8	
Summe:	80	
Note:		

**Bepunktung Multiple Choice:** Geben Sie an, welche Antworten zu folgenden Fragen zutreffen, indem Sie die Ziffern der richtigen Antworten notieren. Alle nicht notierten Ziffern sind automatisch als falsch klassifiziert (Sie können sich also nicht enthalten). Beispiel: Angenommen, die Antworten 1,2,3 sind korrekt, 4,5,6 sind falsch, so notieren Sie in der Antwort-Datei 1,2,3 (Antworten 4,5,6 werden *NICHT* notiert und sind damit automatisch als falsch gekennzeichnet).

Pro richtiger Antwort bekommen Sie einen halben Punkt. Die Punkte werden pro Aufgabenblock abgerundet.

**Aufgabe 1 Allgemeine Fragen**

(18 Punkte)

(a) Gegeben sei ein vorsortiertes Array der Länge  $n$ . Welche der folgenden Methoden ist/sind hierfür geeignete Suchalgorithmen?

- 1 Lineare Suche
- 2 Binäre Suche
- 3 Tiefensuche
- 4 Interpolationssuche
- 5 Breitensuche
- 6 Keines der Verfahren

(b) Welche der folgenden Aussagen bezüglich AVL-Bäumen sind korrekt?

- 1 Es ist ein ternärer Suchbaum.
- 2 Es ist ein binärer Suchbaum.
- 3 Für alle Knoten gilt, dass die Höhen der beiden Teilbäume sich höchstens um eins unterscheiden.
- 4 An der Wurzel muss der linke sowie der rechte Teilbaum die gleiche Höhe haben.
- 5 Nach jeder Such-Operation muss eventuell rebalanciert werden.
- 6 Beim Löschen von Elementen aus einem AVL-Baum werden mehr als Höhe  $h$  viele Rotationen benötigt.

(c) Welche der folgenden Aussagen bezüglich B-Bäumen sind korrekt?

- 1 B-Bäume dienen dem Ziel der effizienten Nutzung des Primärspeichers.
- 2 Im Gegensatz zum Hashing haben B-Bäume eine statische Struktur und können nicht dynamisch wachsen oder schrumpfen.
- 3 Rotationen beheben einen potentiellen Unterlauf nach einer Löschoperation.
- 4 Die minimale Höhe  $h$  eines B-Baumes ist logarithmisch abhängig von der Anzahl der Schlüssel  $s$ .
- 5 Innerhalb eines Knotens (=Seite) sind die Schlüssel bzgl. der auf ihnen definierten Ordnung sortiert.
- 6 B-Bäume sind Mehrwegsuchbäume.

(d) Welche der folgenden Aussagen bezüglich R-Bäumen sind korrekt?

- 1 MBR sind dazu konzipiert ausschließlich Punkte zu beinhalten.
- 2 Gängige Anfragen auf R-Bäumen sind Bereichsanfragen und Nächste-Nachbarn Anfragen.

- 3 Für hochdimensionale Daten sind R-Bäume geeignet da deren Wegweiser kaum überlappen.
- 4 Die Balancierung ist analog zum B-Baum.
- 5 MBRs können auch überlappen.
- 6 Die Distanz zu den MBRs wird abgeschätzt über die maximale durchschnittliche Distanz zwischen Punkt und MBR.

(e) Welche der folgenden Aussagen bezüglich Backtracking sind korrekt?

- 1 Probleme werden nach dem “trial-and-error”-Prinzip gelöst.
- 2 Eine Voraussetzung ist, dass die Lösung sich aus Teillösungen bzw. Komponenten zusammen setzen lässt.
- 3 Eine Voraussetzung ist, dass die Teillösungen getestet werden können.
- 4 Kann eine Teillösung nicht zu einer Gesamtlösung erweitert werden, wird der letzte Schritt rückgängig gemacht, und weitere alternative Schritte werden probiert.
- 5 Eine Voraussetzung ist, dass es für jede Komponente mehrere Wahlmöglichkeiten gibt.
- 6 Geeignete Probleme sind z.B. das Damenproblem, Sudoku.

(f) Welche der folgenden Aussagen bezüglich Branch-and-Bound sind korrekt?

- 1 Bei Problemen, bei denen keine effiziente Verfahren zu deren Lösung bekannt sind, wird Branch-and-Bound als Methode zur Lösung von Optimierungsproblemen eingesetzt.
- 2 Das Paradigma setzt auf einen “trial-and-error” Ansatz.
- 3 Teillösungen werden dann verworfen, wenn das Optimum erreicht werden kann.
- 4 Das Travelling Salesman Problem kann nicht mittels branch-and-bound gelöst werden.
- 5 Für Teillösungen werden Schranken berechnet.
- 6 Beim “branch-and-bound” Paradigma gibt es keine schrittweise Annäherung an eine Lösung sondern eine sog. “one-time” Berechnung.

**Aufgabe 2    Gemischte Aufgaben**

(15 Punkte)

(a) Welches ist der korrekte Bitvektor für die Menge  $M = \{0, 5, 8\}$

i. Gegeben das Universum  $U_1 = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$

1 (0,1,1,0,1,0,1,1,0)

2 (0,1,1,0,0,0,0,0,0)

3 (0,0,1,1)

4 (1,1,1,0)

5 (1,0,0,0,0,1,0,0,1)

6 (1,1,1,1,1,0,0,0,0)

ii. Gegeben das Universum  $U_2 = \{0, 5, 6, 7, 8\}$

1 (1,0,0,0,1)

2 (1,0,1,0,1)

3 (1,1,0,0,1)

4 (0,0,1,1,1)

5 (0,1,0,1,0)

6 (1,0,1,1,1)

(b) Kreuzen Sie die korrekte Edit-Distanz zwischen den Wörtern “Marabu” und “Mammut” an.

1

2

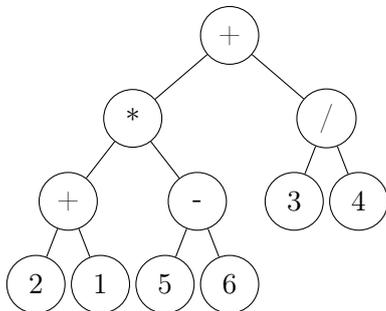
3

4

5

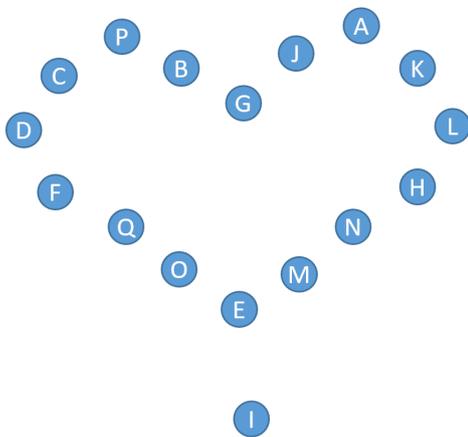
6

- (c) Gegeben sei der folgende Baum. Kreuzen Sie die korrekte Infix- und Präfixschreibweise des Baums bei einem Tiefendurchlauf an.



- 1  $2+1*5-6+3/4$
- 2  $21+56-*34/+$
- 3  $+*/+-342156$
- 4  $/34+*-56+21$
- 5  $+*+21-56/34$
- 6  $3/4+5-6*2+1$

- (d) Welche Folge(n) entsprechen der konvexen Hülle folgender Punktemenge.

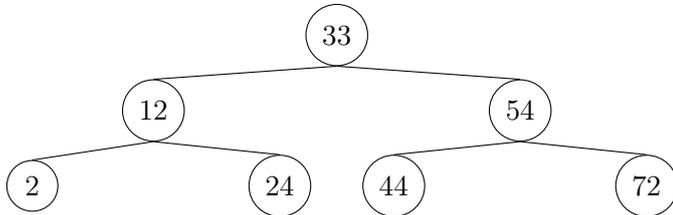


- 1 A,K,L,H,N,M,I,O,Q,F,D,C,P,B,G,J
- 2 A,P,C,D,F,I,H,L,K
- 3 A,K,L,H,I,F,D,C,P
- 4 L,H,F,D,A,K
- 5 A,P,C,D,F,Q,O,E,M,N,H,L,K
- 6 A,K,L,M,I,E,O,Q,F,D,C,P,B,G,J

**Aufgabe 3 Suchbäume**

(9 Punkte)

(a) Gegeben sei der folgende balancierte Fibonacci-Baum.



Löschen Sie nun die Knoten 2 und 24 und fügen Sie die 36 hinzu.

i. Kreuzen Sie die resultierende Rotationsart an, die vorgenommen werden muss, um den Baum zu balancieren.

1 Rechtsrotation

2 Linksrotation

3 LR-Rotation

4 RL-Rotation

ii. Nehmen Sie die entsprechende Rotation vor und kreuzen Sie den korrekten Baum (Präfix-, Infix- oder Postfixschreibweise) an, der dabei entsteht.

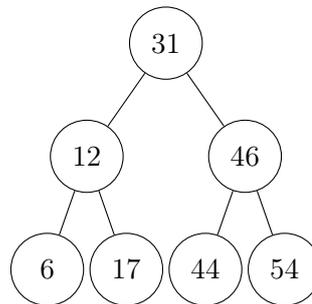
1 44,33,12,36,54,72

2 12,33,36,44,54,72

3 12,36,33,72,54,44

4 44,33,54,12,36,72

(b) Gegeben ist der folgende binäre Suchbaum.



Kreuzen Sie die gültige(n) Reihenfolge(n) an, in der die Werte eingefügt werden müssen, damit der oben abgebildete Baum entsteht. Starten Sie dabei mit einem leeren Baum.

1 31 12 6 17 46 44 54

2 31 12 17 6 46 54 44

3 31 46 44 54 12 6 17

4 31 12 44 54 46 6 17

5 31 46 54 44 12 17 6

6 31 12 46 6 17 44 54

(c) Angenommen, Sie suchen den Wert  $x$  in einem binären Suchbaum. Gegeben ist die folgende Sequenz von bereits untersuchten Schlüsselwerten:

17 144 78 20 62 60 24

Geben Sie die unterste und oberste Grenze an, den der Wert  $x$  annehmen kann.

1 25 und 59

2 21 und 59

3 24 und 24

4 19 und 63

**Aufgabe 4 Graphalgorithmen**

(16 Punkte)

(a) Gegeben sei die folgende Adjazenzmatrix. Welche Aussagen über den zugehörigen Graphen treffen zu?

0	0	1	0	1	0
1	0	0	1	0	0
1	1	0	0	0	0
1	0	0	0	0	1
0	0	1	1	0	0
0	0	0	0	1	1

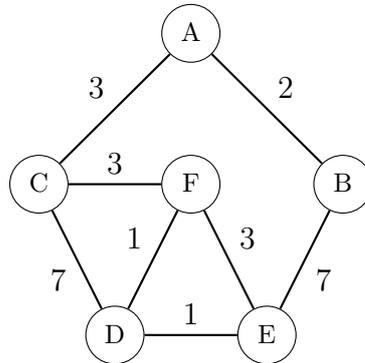
- 1 Für alle Knotenpaare  $(v, w) \in E$  gilt:  $v \neq w$ .
- 2 Für alle Knotenpaare existiert ein kürzester Pfad.
- 3 Zwischen allen Knotenpaaren existiert ein Pfad.
- 4 Der Graph ist ein Baum.
- 5 Der Graph ist planar (lässt sich zeichnen, ohne dass sich Kanten kreuzen).
- 6 Die Matrix repräsentiert einen ungerichteten Graphen.
- 7 Genau ein Knoten hat Knotengrad 3.
- 8 Jeder Pfad im Graphen ist einfach.

(b) Gegeben sei die folgende Adjazenzliste. Welche Aussagen über den zugehörigen Graphen treffen zu?

$A \rightarrow B$   
 $B \rightarrow A, D, F$   
 $C \rightarrow F$   
 $D \rightarrow B, E, G$   
 $E \rightarrow D$   
 $F \rightarrow B, C$   
 $G \rightarrow D$

- 1 Für alle Knotenpaare  $(v, w) \in E$  gilt:  $v \neq w$ .
- 2 Der minimale Spannbaum des Graphen besitzt weniger Kanten.
- 3 Es gibt Knotenpaare ohne kürzesten Pfad.
- 4 Der Graph enthält mindestens einen Zykel.
- 5 Die Adjazenzliste repräsentiert einen ungerichteten Graphen.
- 6 Der Graph ist ein Baum.
- 7 Es gibt genau einen kürzesten Pfad zwischen  $E$  und  $C$ .
- 8 Genau ein Knoten hat Knotengrad 2.

Gegeben sei der folgende Graph  $G$ :



(c) Wenden Sie den Dijkstra-Algorithmus mit Startknoten  $A$  auf  $G$  an. Welche Aussagen treffen zu?

- 1  $B$  wird direkt nach  $A$  betrachtet.
- 2  $C$  wird direkt nach  $A$  betrachtet.
- 3  $E$  wird direkt nach  $B$  bearbeitet.
- 4  $E$  wird direkt nach  $D$  bearbeitet.
- 5  $D$  wird direkt nach  $F$  bearbeitet.
- 6  $C$  wird direkt nach  $E$  bearbeitet.
- 7 Der kürzeste Pfad von  $A$  nach  $D$  hat Distanz 7.
- 8 Der kürzeste Pfad von  $A$  nach  $D$  hat Distanz 10.
- 9 Der kürzeste Pfad von  $A$  nach  $E$  hat Distanz 9.
- 10 Der kürzeste Pfad von  $A$  nach  $E$  hat Distanz 8.

(d) Sei  $G'$  der minimale Spannbaum zu  $G$ . Welche Aussagen treffen zu?

- 1 Die Algorithmen von Kruskal und Prim liefern identische Ergebnisse.
- 2  $G'$  besitzt genau zwei Kanten weniger als  $G$ .
- 3  $G'$  ist nicht eindeutig.
- 4 Die Summe der Kantengewichte von  $G'$  ist 9.
- 5  $G'$  enthält nur Kanten mit Gewicht kleiner 4.
- 6 Der größte vorkommende Knotengrad in  $G'$  ist 2.

**Aufgabe 5    Algorithmenanalyse**

(9 Punkte)

Die folgenden Algorithmen wurden erstellt, um den Median einer Liste von ganzen Zahlen zu bestimmen. Sie dürfen annehmen, dass nur nicht-leere Listen mit positiven ganzen Zahlen übergeben werden. Kommen zwei Zahlen als Median in Frage (Liste hat gerade Länge), dann nehmen wir den Mittelwert beider Zahlen.

(a) Welche Aussagen treffen auf folgenden Algorithmus zu?

```
def M1(x):
    while len(x) > 2:
        min = infinity
        max = -infinity
        for i in x:
            if x > max:
                max = x
            if x < min:
                min = x:
        x.remove(min)
        x.remove(max)
    if len(x) == 2:
        return (x[0]+x[1])/2
    return x[0]
```

- 1  $M1$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n)$
- 2  $M1$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n \log n)$
- 3  $M1$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n^2)$
- 4  $M1$  terminiert.
- 5  $M1$  ist determiniert.
- 6  $M1$  ist ein rekursiver Algorithmus.

(b) Welche Aussagen treffen auf folgenden Algorithmus zu?

```
def M2(x):
    x = mergesort(x)
    if len(x) % 2 == 0: // % entspricht der Modulo-Operation
        return (x[len(x)/2+1]+x[len(x)/2])/2
    else:
        return x[len(x)/2]
```

- 1  $M2$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n)$
- 2  $M2$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n \log n)$
- 3  $M2$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n^2)$
- 4  $M2$  ist partiell korrekt.
- 5  $M2$  ist deterministisch.
- 6  $M2$  ist endrekursiv.

(c) Welche Aussagen treffen auf folgenden Algorithmus zu?

```
def M3(x):
    n = len(x)
    if n == 1
        return x[0]
    if n == 2
        return (x[0]+x[1])/2
    // drei unterschiedliche (no-repeat) Elemente
    // zufaellig auswaehlen:
    x1, x2, x3 = x.pick_random(3, no-repeat)
    if x1 < x2 < x3:
        x.remove(x1)
        x.remove(x3)
    return M3(x)
```

- 1  $M3$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n)$
- 2  $M3$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n \log n)$
- 3  $M3$  hat im Worst-Case eine Komplexität von  $\mathcal{O}(n^2)$
- 4  $M3$  ist deterministisch.
- 5  $M3$  ist partiell korrekt.
- 6  $M3$  ist endrekursiv.

**Aufgabe 6    Sortieralgorithmen**

(5 Punkte)

(a) Gegeben sei das folgende Array

[5, 10, 1, 8, 9, 7, 20, 3, 6, 4, 11, 18, 14, 16, 19, 2, 13, 17, 15, 12].

und folgende Zwischenergebnisse des einfachen Sortieralgorithmus. Gemäß der Vorlesung, welcher Algorithmus wurde jeweils am wahrscheinlichsten benutzt?

i. [1, 5, 3, 6, 4, 7, 8, 9, 10, 11, 2, 13, 14, 15, 12, 16, 17, 18, 19, 20 ]

1 MergeSort

2 BubbleSort

3 InsertionSort

4 SelectionSort

ii. [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 14, 18, 11, 16, 19, 20, 13, 17, 15, 12]

1 MergeSort

2 BubbleSort

3 InsertionSort

4 SelectionSort

iii. [1, 3, 4, 5, 6, 7, 8, 9, 10, 14, 18, 20, 11, 16, 19, 2, 13, 17, 15, 12]

1 MergeSort

2 BubbleSort

3 InsertionSort

4 SelectionSort

- (b) Ein Array wird schrittweise mit einem Sortieralgorithmus der Vorlesung sortiert:  
[5, 10, 1, 8, 9, 7, 20, 3, 6, 4, 11, 18, 14, 16, 19, 2, 13, 17, 15, 12]

1. Schritt: [5, 10, 1, 8, 9, 7, 20, 3, 6, 4, 11, 18, 14, 16, 19, 2, 13, 15, 17, 12]
2. Schritt: [5, 10, 1, 8, 9, 7, 20, 3, 4, 6, 11, 18, 14, 16, 19, 2, 13, 12, 15, 17]
3. Schritt: [1, 5, 8, 9, 10, 3, 4, 6, 7, 20, 11, 14, 16, 18, 19, 2, 12, 13, 15, 17]
4. Schritt: [1, 3, 4, 5, 6, 7, 8, 9, 10, 20, 2, 11, 12, 13, 14, 15, 16, 17, 18, 19]
5. Schritt: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

Was ist die Worst-Case-Laufzeit dieses Algorithmus bei einem Array der Länge  $n$ ?

- 1  $O(n \cdot \log(n))$
- 2  $O(n^{-2})$
- 3  $O(n)$
- 4  $O(\log(n))$

- (c) Was ist die größte untere Schranke der Anzahl der benötigten Vergleiche in jedem vergleichsbasierten Sortierverfahren im Worst-Case, mit  $n$  als Anzahl der zu sortierenden Objekte?

- 1  $O(n \cdot \log(n))$
- 2  $O(n^2)$
- 3  $O(n)$
- 4  $O(\log(n))$

**Aufgabe 7 Hashing**

(8 Punkte)

- (a) Tragen Sie die folgenden Schlüssel in der angegebenen Reihenfolge nach geschlossenem Hashing mit offener Addressierung in eine Hashtabelle der Größe  $m = 10$  ein. Verwenden Sie die Hashfunktion  $h(x) = x \bmod 10$  und achten Sie darauf für jeden Schlüssel, wie oft sondiert werden muss.

5, 24, 4, 11, 15, 16, 26, 2

Nutzen Sie dafür einmal lineares Sondieren und einmal quadratisches Sondieren. Welche Aussagen sind nach Anwendung des jeweils angegebenen Prinzips richtig?

i. Bezüglich linearem Sondieren:

- 1 Bei der Index 0 und Index 3 stehen keine Elemente.
- 2 Insgesamt sind 8 Sondierungen notwendig
- 3 Insgesamt sind 4 Sondierungen notwendig
- 4 Bei Index 9 wird "26" eingetragen.

ii. Bezüglich quadratischem Sondieren:

- 1 Bei der Index 0 und Index 3 stehen keine Elemente.
- 2 Insgesamt sind 6 Sondierungen notwendig
- 3 Insgesamt sind 4 Sondierungen notwendig
- 4 Bei linearem Sondieren sind mehr Sondierungen notwendig als bei quadratischem Sondieren

(b) Gegeben ist folgende Reihenfolge an Schlüsseln:

4, 29, 5, 11, 19, 17, 26, 2

und eine Hashtabelle der Größe  $m = 3$  und die Hashfunktion  $h(x) = x \bmod 3$ . Tragen Sie oben genannten Schlüssel in der angegebenen Reihenfolge nach dem Prinzip des **Offenes Hashings** ein. Nutzen Sie als sekundäre Datenstruktur ein Array. Welche Aussagen sind nach Anwendung des Verfahrens richtig?

Index	
0	
1	
2	

- 1 Bei Index 0 hat das Array die Länge 2
- 2 Bei Index 0 hat das Array keine Elemente
- 3 Insgesamt sind 4 Sondierungen notwendig
- 4 Bei Index 1 steht das Array [4, 19, 26]

(c) Welche Liste(n) von ganzen Zahlen ist/sind geeignet, sodass die Hashfunktion  $h(x) = x \bmod 5$  eine perfekte Hashfunktion ist.

- 1 [1,2,3,7,9]
- 2 [5,6,7,8,9]
- 3 [5,10,15,20,25]
- 4 [123,345,356]