

Informatik II

Übung 8

Giuseppe Accaputo, Felix Friedrich, Patrick Gruntz, Tobias Klenze, Max Rossmannek, David Sidler, Thilo Weghorn

FS 2017

Heutiges Programm

- 1 Hashtabellen
- 2 Wiederholung Verkettete Listen
- 3 Offene Hashverfahren

Aus aktuellem Anlass

Der **vollständige Code zu den Programmen der Vorlesung ist auf der Homepage der Vorlesung** verlinkt. [Umfrage hat ergeben, dass einige das nicht wussten]

Die **Lösungen zu den Clickerfragen werden immer auf der Homepage der Vorlesung veröffentlicht** [Umfrage hat ergeben, dass einige das nicht wussten]

Rechenregeln Modulo

$$(a + b) \bmod m = ((a \bmod m) + (b \bmod m)) \bmod m$$

$$(a \cdot b) \bmod m = ((a \bmod m) \cdot (b \bmod m)) \bmod m$$

Aufgabe: Berechnen Sie

$$12746357 \bmod 11$$

Rechenregeln Modulo

Aufgabe: Berechnen Sie

$$12746357 \bmod 11$$

Rechenregeln Modulo

Aufgabe: Berechnen Sie

$$12746357 \bmod 11$$

$$= (7 + 5 \cdot 10 + 3 \cdot 10^2 + 6 \cdot 10^3 + 4 \cdot 10^4 + 7 \cdot 10^5 + 2 \cdot 10^6 + 1 \cdot 10^7) \bmod 11$$

Rechenregeln Modulo

Aufgabe: Berechnen Sie

$$\begin{aligned} & 12746357 \bmod 11 \\ &= (7 + 5 \cdot 10 + 3 \cdot 10^2 + 6 \cdot 10^3 + 4 \cdot 10^4 + 7 \cdot 10^5 + 2 \cdot 10^6 + 1 \cdot 10^7) \bmod 11 \\ &= (7 + 50 + 3 + 60 + 4 + 70 + 2 + 10) \bmod 11 \end{aligned}$$

Für die zweite Gleichheit haben wir verwendet, dass $10^2 \bmod 11 = 1$.

Rechenregeln Modulo

Aufgabe: Berechnen Sie

$$\begin{aligned} & 12746357 \bmod 11 \\ &= (7 + 5 \cdot 10 + 3 \cdot 10^2 + 6 \cdot 10^3 + 4 \cdot 10^4 + 7 \cdot 10^5 + 2 \cdot 10^6 + 1 \cdot 10^7) \bmod 11 \\ &= (7 + 50 + 3 + 60 + 4 + 70 + 2 + 10) \bmod 11 \\ &= (7 + 6 + 3 + 5 + 4 + 4 + 2 + 10) \bmod 11 \end{aligned}$$

Für die zweite Gleichheit haben wir verwendet, dass $10^2 \bmod 11 = 1$.

Rechenregeln Modulo

Aufgabe: Berechnen Sie

$$\begin{aligned} & 12746357 \bmod 11 \\ &= (7 + 5 \cdot 10 + 3 \cdot 10^2 + 6 \cdot 10^3 + 4 \cdot 10^4 + 7 \cdot 10^5 + 2 \cdot 10^6 + 1 \cdot 10^7) \bmod 11 \\ &= (7 + 50 + 3 + 60 + 4 + 70 + 2 + 10) \bmod 11 \\ &= (7 + 6 + 3 + 5 + 4 + 4 + 2 + 10) \bmod 11 \\ &= 10 \bmod 11. \end{aligned}$$

Für die zweite Gleichheit haben wir verwendet, dass $10^2 \bmod 11 = 1$.

Implementation Hashfunktion (String) in Java

$$h_{b,m}(s) = \left(\sum_{i=0}^{l-1} s_i \cdot b^i \right) \bmod m$$

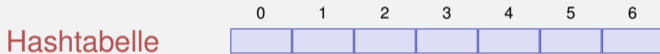
```
int ComputeHash(int m, String s) {
    int sum = 0;
    int b = 1;
    for (int k = 0; k < s.length(); ++k) {
        sum = (sum + s.charAt(k) * b) % m;
        b = (b * 31) % m;
    }
    return sum;
}
```

Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12

Direkte Verkettung der Überläufer



Überläufer

Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer

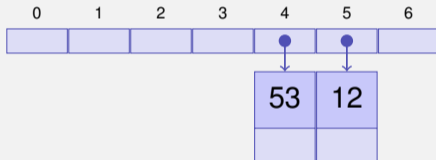
Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer

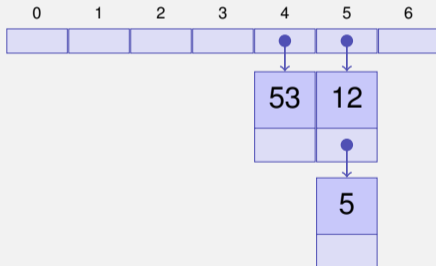
Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer

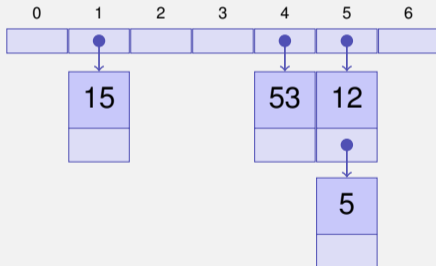
Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15, 2

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer

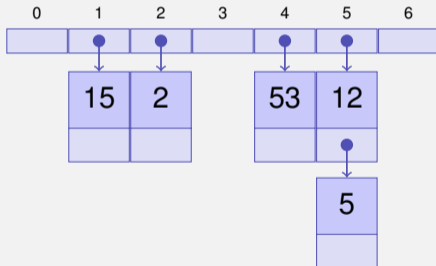
Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15, 2, 19

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer

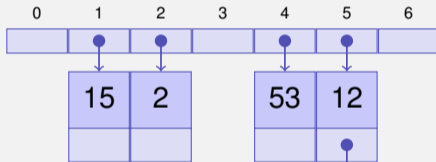
Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

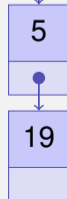
Schlüssel 12, 53, 5, 15, 2, 19, 43

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer



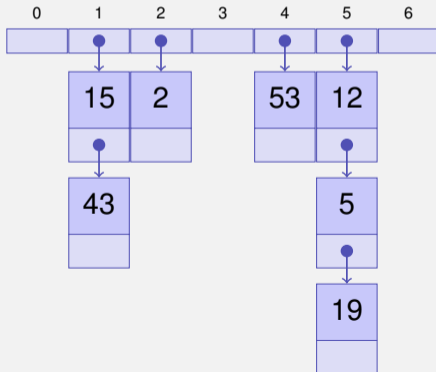
Behandlung von Kollisionen

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15, 2, 19, 43

Direkte Verkettung der Überläufer

Hashtabelle



Überläufer

Algorithmen zum Hashing mit Verkettung

- **contains**(k) Durchsuche Liste an Position $h(k)$ nach k . Gib wahr zurück, wenn gefunden, sonst falsch.
- **put**(k) Prüfe ob k in Liste an Position $h(k)$. Falls nein, füge k am Ende der Liste ein. Andernfalls Fehlermeldung.
- **get**(k) Prüfe ob k in Liste an Position $h(k)$. Falls ja, gib die Daten zum Schlüssel k zurück. Andernfalls Fehlermeldung
- **remove**(k) Durchsuche die Liste an Position $h(k)$ nach k . Wenn Suche erfolgreich, entferne das entsprechende Listenelement.

Komponenten der Hashtabelle

- Hashfunktion
- ~~Probing (nur beim offenen Hashing)~~
- Operationen auf der verketteten Liste (beim geschlossenen Hashing)
 - Einfügen
 - Suchen
 - Löschen

Verkettete Listen

Gegeben:

```
public class ListNode {
    String key;
    int value;
    ListNode next;

    ListNode (String k, int val, ListNode nxt) {
        key = k;
        value = val;
        next = nxt;
    }
}
```

Einfügen

```
public class ListNode {  
    String key;  
    int value;  
    ListNode next;  
  
    ListNode (String k, int val, ListNode nxt) {...}  
}
```

Ergänzen Sie

```
// Return the new root of a list containing new  
// root is the current root of the list  
ListNode Insert(ListNode root, String key, int value){  
  
    .....  
  
}
```

Einfügen

```
public class ListNode {  
    String key;  
    int value;  
    ListNode next;  
  
    ListNode (String k, int val, ListNode nxt) {...}  
}
```

Ergänzen Sie

```
// Return the new root of a list containing new  
// root is the current root of the list  
ListNode Insert(ListNode root, String key, int value){  
  
    return new ListNode(k, key, root);  
  
}
```

Suchen

```
public class ListNode {  
    String key;  
    int value;  
    ListNode next;  
}
```

Ergänzen Sie

```
// pre: root is the current root of the list  
// post: return if list contains element with key  
boolean Contains(ListNode root, String key){  
    .....  
    .....  
    .....  
}
```


Suchen

```
// pre: root is the current root of the list
// post: return if list contains element with key
boolean Contains(ListNode root, String key){
    ListNode node = root;
    while (node != null & !key.equals(node.key)){
        node = node.next;
    }
    return node != null;
}
```

Finden

```
public class ListNode {  
    String key;  
    int value;  
    ListNode next;  
}
```

Ergänzen Sie

```
// pre: key is contained in the list,  
//      root is the current root of the list  
// post: return element containing key from the list  
ListNode Get(ListNode root, String key){  
    .....  
    .....  
    .....  
}
```

Finden

```
// pre: key is contained in the list,  
//      root is the current root of the list  
// post: return element containing key from the list  
ListNode Get(ListNode root, String key){  
    ListNode result = root;  
    while (!key.equals(result.key)){  
        result = result.next;  
    }  
    return result;  
}
```

Löschen

```
public class ListNode {  
    String key;  
    int value;  
    ListNode next;  
}
```

Ergänzen Sie

```
// pre: key is contained in the list,  
//      root is the current root of the list  
// post: remove element containing key from the list  
//      and return the new root of the list  
ListNode Remove(ListNode root, String key){  
    .....  
    .....  
    .....  
}
```

Löschen

```
// pre: key is contained in the list,  
//      root is the current root of the list  
// post: remove element containing key from the list  
//      and return the new root of the list  
ListNode Remove(ListNode root, String key){  
    if (key.equals(root.key)){  
        return root.next;  
    }  
    Node prev = root;  
    while (!key.equals(prev.next.key)){  
        prev = prev.next;  
    }  
    prev.next = prev.next.next;  
    return root;  
}
```

Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

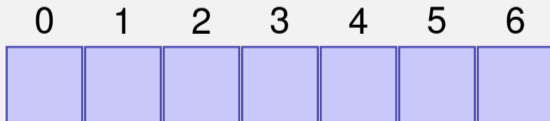
Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12



Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53

0	1	2	3	4	5	6
					12	

Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5

0	1	2	3	4	5	6
				53	12	

Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15

0	1	2	3	4	5	6
			5	53	12	

Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15, 2

0	1	2	3	4	5	6
	15		5	53	12	

Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15, 2, 19

0	1	2	3	4	5	6
	15	2	5	53	12	

Lineares Sondieren

$$s(j, k) = j \Rightarrow$$

$$S(k) = (h(k) \bmod m, (h(k) - 1) \bmod m, \dots, (h(k) + 1) \bmod m)$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \bmod m$.

Schlüssel 12, 53, 5, 15, 2, 19

0	1	2	3	4	5	6
19	15	2	5	53	12	

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod{m}$$

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12

0	1	2	3	4	5	6

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12, 53

0	1	2	3	4	5	6
					12	

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12, 53, 5

0	1	2	3	4	5	6
				53	12	

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12, 53, 5, 15

0	1	2	3	4	5	6
				53	12	5

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12, 53, 5, 15, 2

0	1	2	3	4	5	6
	15			53	12	5

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12, 53, 5, 15, 2, 19

0	1	2	3	4	5	6
	15	2		53	12	5

Quadratisches Sondieren

$$s(j, k) = \lceil j/2 \rceil^2 (-1)^j$$

$$S(k) = (h(k) + 1, h(k) - 1, h(k) + 4, h(k) - 4, \dots) \pmod m$$

Beispiel $m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod m$.

Schlüssel 12, 53, 5, 15, 2, 19

0	1	2	3	4	5	6
19	15	2		53	12	5
.

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m - 1)h'(k)) \pmod m$

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m-1)h'(k)) \pmod{m}$$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod{7}$, $h'(k) = 1 + k \pmod{5}$.

Schlüssel 12

0	1	2	3	4	5	6

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m - 1)h'(k)) \pmod m$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod 7$, $h'(k) = 1 + k \pmod 5$.

Schlüssel 12, 53

0	1	2	3	4	5	6
					12	

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m-1)h'(k)) \pmod{m}$$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod{7}$, $h'(k) = 1 + k \pmod{5}$.

Schlüssel 12, 53, 5

0	1	2	3	4	5	6
				53	12	

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m-1)h'(k)) \pmod{m}$$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod{7}$, $h'(k) = 1 + k \pmod{5}$.

Schlüssel 12, 53, 5, 15

0	1	2	3	4	5	6
			5	53	12	

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m-1)h'(k)) \pmod{m}$$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod{7}$, $h'(k) = 1 + k \pmod{5}$.

Schlüssel 12, 53, 5, 15, 2

0	1	2	3	4	5	6
	15		5	53	12	

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m-1)h'(k)) \pmod{m}$$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod{7}$, $h'(k) = 1 + k \pmod{5}$.

Schlüssel 12, 53, 5, 15, 2, 19

0	1	2	3	4	5	6
	15	2	5	53	12	

Double Hashing

Zwei Hashfunktionen $h(k)$ und $h'(k)$. $s(j, k) = j \cdot h'(k)$.

$S(k) = (h(k) - h'(k), h(k) - 2h'(k), \dots, h(k) - (m - 1)h'(k)) \pmod m$

Beispiel:

$m = 7$, $\mathcal{K} = \{0, \dots, 500\}$, $h(k) = k \pmod 7$, $h'(k) = 1 + k \pmod 5$.

Schlüssel 12, 53, 5, 15, 2, 19

0	1	2	3	4	5	6
19	15	2	5	53	12	

Fragen oder Anregungen?