Blatt 4.

Aufgabe 1

H - Haschtabelle, $m = 13$, $h_1, h_2, h_3 : \mathbb{N}_0 \mapsto \{0, \dots, m-1\}$
Hashfunktionen definiert wie folgt:

- $h_1 := \overline{X} \mod m$
- $h_2 := 3X \mod m$
- $h_3 := X + 1 \mod m$

a) Lineares Sondieren unter der Benutzung von h_1.

---

1) 23

$h_1(23) = (2+3) \mod 13 = 5$

0	1	2	3	4	5	6	7	8	9	10	11	12
					23							

---

2) 12

$h(12) = (1+2) \mod 13 = 3$

0	1	2	3	4	5	6	7	8	9	10	11	12
			12		23							

---

3) 75

$h(75) = (7+5) \mod 13 = 12$

0	1	2	3	4	5	6	7	8	9	10	11	12
			12		23							75

0	1	2	3	4	5	6	7	8	9	10	11	12
			12		23	945						75

4) 945

$h(945) = (9+4+5) \mod 13 = 5$

Kollision

$\Rightarrow h(945, 1) = (h(945) + 1) \mod 13 = 6 \mod 13 = 6$

0	1	2	3	4	5	6	7	8	9	10	11	12
			12	30	23	945						75

5) 30

$h(30) = (3+0) \bmod 13 = 3$

Kollision

$\Rightarrow h(30,1) = (h(30)+1) \bmod 13 = 4$

0	1	2	3	4	5	6	7	8	9	10	11	12
			12	30	23	945	99					75

6) 99

$h(99) = (9+9) \bmod 13 = 5$

Kollision

$\Rightarrow h(99,1) = (h(99)+1) \bmod 13 = 6$

Kollision

$\Rightarrow h(99,2) = (h(99)+2) \bmod 13 = 7$

0	1	2	3	4	5	6	7	8	9	10	11	12
345			12	30	23	945	99					75

7) 345

$h(345) = (3+4+5) \bmod 13 = 12$

Kollision

$\Rightarrow h(345,1) = (h(345)+1) \bmod 13 = 0$

---

b) Doppel-Hasching unter Benutzung von h_2 und h_3.

0	1	2	3	4	5	6	7	8	9	10	11	12
				23								

1) 23

$h(23) = (h_2(23) + 0 \cdot h_3(23)) \bmod 13 = 4$

2) 12

0	1	2	3	4	5	6	7	8	9	10	11	12
				23						12		

$$h(23) = (h_2(12) + 1 \cdot h_3(12)) \bmod 13 = 10$$

---

3) 75

0	1	2	3	4	5	6	7	8	9	10	11	12
75				23						12		

$$h(75) = (h_2(75) + 2 \cdot h_3(75)) \bmod 13 = 0$$

---

4) 945

0	1	2	3	4	5	6	7	8	9	10	11	12
75				23	945					12		

$$h(945) = (h_2(945) + 3 \cdot h_3(945)) \bmod 13 = 6$$

---

5) 30

0	1	2	3	4	5	6	7	8	9	10	11	12
75				23	945	30				12		

$$h(30) = (h_2(30) + 4 \cdot h_3(30)) \bmod 13 = 6$$

---

6) 99

0	1	2	3	4	5	6	7	8	9	10	11	12
75				23	945	30			99	12		

$$h(99) = (h_2(99) + 5 \cdot h_3(99)) \bmod 13 = 6$$

$\xrightarrow{\text{Kollision}}$ $h(99) = (h_2(99) + 6 \cdot h_3(99)) \bmod 13 = 0$

$\xrightarrow{\text{Kollision}}$ $h(99) = (h_2(99) + 7 \cdot h_3(99)) \bmod 13 = 9$

---

7) 345

0	1	2	3	4	5	6	7	8	9	10	11	12
75				23	945	30	345		99	12		

$$h(345) = (h_2(345) + 8 \cdot h_3(345)) \bmod 13 = 7$$

2) $|S| \geq y \cdot m$ $\quad |Y| \geq y$

zZ.: $h(x_1) = h(x_2)$ für $\forall x_1, x_2 \in Y$

angenommen h ist eine zufällige Hashfunktion so ist die Wahrscheinlichkeit für die Berechnung eines Schlüssels $\frac{1}{m}$.

Multipliziert mit der Anzahl an Elementen $|S| \geq y \cdot m$ erhält man die Anzahl der Elemente pro Schlüssel $|Y| \geq y$

$$\rightarrow \quad y \cdot m \cdot \frac{1}{m} = y$$

$\Rightarrow$ wenn $x_1, x_2 \in Y$ so gilt, dass die Hashfunktion für beide Werte den selben Schlüssel errechnet. Somit gilt: $h(x_1) = h(x_2)$

b) für die Laufzeit gilt $T(n) \in O(1 + \alpha)$

mit $\alpha = \frac{n}{m}$ und $n = y \cdot m$

$$\alpha = \frac{y \cdot m}{m} = y$$

$$\Rightarrow T(n) \in O(1 + y)$$

da $\quad O(1 + y) \in O(y)$

$$T(n) \in O(y)$$

3) a) Der Algorithmus berechnet den Betrag der Differenz einer Stelle eines Arrays mit den bereits durchlaufenen Stellen des Arrays und gibt true aus, wenn der Betrag dieser Differenz gleich groß ist wie ein Eintrag in dem gesamten Array

Laufzeit: $(n-1) \cdot n \cdot n = n^3 - n^2$
$\leq n^3 \leq c \cdot n^3$

$T(n) \in \mathcal{O}(n^3)$

b) Alternativer Algorithmus: $\mathcal{O}(n^2)$

H : Hashtabelle

```
• for i=1 to n-1 do:
•     for j=0 to i-1 do:
•         insert in H = |A[i] - A[j]|

• for k=0 to n-1 do:
•     if A[k] in H:
•         return true
• return false
```

c) Alternativer Algorithmus: $\mathcal{O}(n^2 \log n)$

mit BinSearch(Array, x)

```
· Sort(A)            # z.Bsp.: MergeSort ∈ O(n log n)
· for i := 1 to n-1 do:
·     for j := 0 to i-1 do:
·         if ( BinSearch (A, |A[i] - A[j]|) ):
·             return true
· return false
```