

Algodat - Rep. II

• \Rightarrow für Unterpunkte bei einer Aufgabe

① Greedy Algo. Frage von Rupert Schöff aus dem Tuel Forum \rightarrow A2 & A3

(A3) \Rightarrow Ob mit Prim/Kruskal gelöst ist egal / Test 2, SS 2016

a) • 1-2-3-4-4 Anzahl = 1

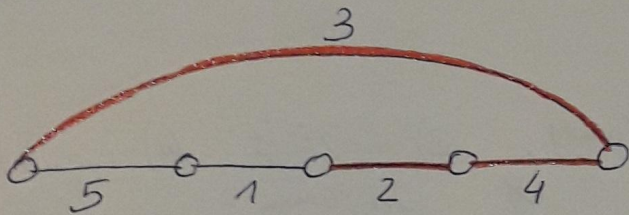
• 1-2-3-4-4 Anzahl = 3 (man kann hier auswählen, welche 4 man wählt, daher mehr Möglichkeiten)

• 1-2-3 Anzahl = 1

b) •



\Rightarrow Schwachstellen eines Algo. finden & diese ausnutzen, hier wählt Kruskal 1-2-3, Kanten hängen jedoch NICHT zusammen \Rightarrow KEIN Spannbaum



\Rightarrow nicht korrekt, da das kleinste Kantengewicht (1) nicht enthalten ist

\Rightarrow man muss die Gewichte so wählen, dass der Algo. normal zur kleinsten Kante gelangen kann

(A2) \Rightarrow Nachträgest

1) $L = 2, 3, 1$

$\swarrow \searrow$
5, 1

$\swarrow \searrow$
6 (Aufwand = 6)

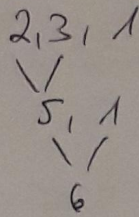
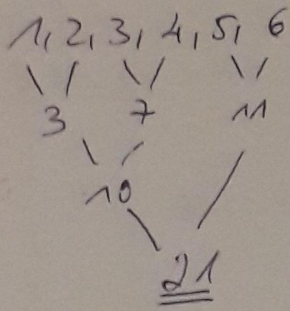
$\Rightarrow 5+6 = \underline{\underline{11}}$ Aufwand

Wenn man jedoch umsortiert z.B. aufsteigend sortiert, dann erhält man ein weitaus besseres Ergebnis, d.h. beim Algo. fehlt das Sortieren bzw. Bed. das L sortiert sich muss

$\Rightarrow L = 1, 2, 3$
 $\swarrow \searrow$
3, 3
 $\swarrow \searrow$
6

$\Rightarrow 3+6 = \underline{\underline{9}}$ Aufwand

2) nicht optimal



$\Rightarrow 5+6 = \underline{11}$

3) Was sollte man am Algo. ändern?

\Rightarrow While \neq sortiert
 $\}$

... Anfang der Liste
 \hookrightarrow sortiert in die...

Wichtig

Die 2 kleinsten Werte
 sollten immer vorne sein!

SS 2016, Test 2

c)

1) $m = 2^{10}$ (egal)

$A = \frac{\pi}{2} \Rightarrow \checkmark \checkmark$

2) $m = 13$ (egal)

$A = 2$
 \Downarrow
 keine irr. Zahl

3) $m = 2^5$

2er Potenzen
 sind hier
 schlecht

4) $m = 27 \Rightarrow 3^3 = \text{Potenz}$

\Rightarrow Schlecht

$h_1(k) = k \bmod 23$

\Downarrow
 sollte m sein

hier wird nämlich nicht
 die komplette
 Hashfkt. abgedeckt,
 man ist gezwungen
 $h_2(k)$ zu verwenden

Wichtig = hier auch
 Begründung erforderlich!
 sonst 0 Punkte

HÜ Bsp. 28

a) 26209 = \downarrow zu groß

4096 = \downarrow 2er Potenz $\Rightarrow 2^{12}$

3517 = $m \rightarrow$ sehr knapp

4127 = \checkmark (ca. sollte ca. 20% über dem sein was zu speichern ist & Primzahl
 \Rightarrow ideal!

3499 = \downarrow zu klein

c) $h_1(k)$ & $h_2(k)$ sollten teilerfremd sein!

Aus dem Twinkl Discussions Forum

$$h_1(k) = 4 \bmod 13$$

(Wenn etwas unterstrichen ist, bedeutet es, dass dieser Wert an der jeweiligen Stelle neu hinzugefügt wurde)

= 1. Berechnung, hier kommt der zu einfügende Wert

= Werte so zu Beginn in T vorgegeben oder selbstständig einzufügen

0	1	2	3	4	5	6	7	8	9	10	11	12
39		28	29	43 <u>91</u>	44	45		<u>43</u>			50	
<u>91</u>		<u>91</u>		<u>91</u>								

(1) $91 \bmod 13 = 0$

(2) $(91 \bmod 5) + 1 = 1 + 1 = 2$ (2 dazuzählen fürs Weitersehen in der Tabelle $h_2(k)$)

(3) $(39 \bmod 5) + 1 = 4 + 1 = 5$

(4) mit 91 2 Stellen weitergehen

(5) 91 \Rightarrow 2 Stellen weiter

(6) $(28 \bmod 5) + 1 = 3 + 1 = 4$

$(43 \bmod 5) + 1 = 4$

(\downarrow = Kollision)

am Algo. von Brent (siehe Pseudocode dazu in den Folien)

$\Rightarrow j$ = Stelle

k^l = blockiert an der Stelle

Anmerkung

\Rightarrow alles was grün geschrieben wurde, bleibt natürlich nicht stehen sondern gehört was hier nur zum Verständnis

z.B. am obigen Bsp.
 $j = 0$
 while
 $k^l = 39$
 $j_1 = 2$ (2. Stelle)
 $j_2 = 5$

if (Wenn $T(j_2)$ used \rightarrow weitersehen)

$j_2 \neq \text{used} \Rightarrow$ hier einfügen an genau dieser Stelle

$j = 2$

etc. \Rightarrow am obigen Bsp. Punkt 6 würde in den else Zweig kommen)

0	1	2	3	4	5	6	7	8	9	10	11	12
			16	30	18		20	21		36 17	<u>36</u>	
				17 y^k						17 y^k		

$$h_1(a) = k \bmod 13$$

$$h_2(b) = (k \bmod 6) + 1$$

① $17 \bmod 13 = 4$

② $(17 \bmod 6) + 1 = 5 + 1 = 6$ (6 dazurück)

$(30 \bmod 6) + 1 = 0 + 1 = 1$

③ $17 \Rightarrow 6$ weiter gehen

$(36 \bmod 6) + 1 = 0 + 1 = 1$

HÜ 4/Bsp. 27 hier Zahlenfolge eintragen

0	1	2	3	4	5	6	7	8	9	10
	127	35		81 70	27	<u>81</u>		17	4	<u>75</u>
				70 y^k					75 y^k	

$$\Rightarrow 70 \bmod 11 = 4$$

$$\Rightarrow (70 \bmod 5) + 1 = 1 \Rightarrow 1 \text{ Stelle weiter gehen} \Rightarrow 5. \text{ Stelle jedoch } b\text{-test} = y$$

$$(81 \bmod 5) + 1 = 2$$

$$\Rightarrow h_2(4) = 5$$

$$h_2(127) = 3 \Rightarrow 9. \text{ Stelle} \rightarrow y^k$$

$$h_2(75) = 1$$

$$\Rightarrow \text{mit } 81 \text{ um } 2. \text{ Stellen weiter} \Rightarrow y^k$$

Zu Bäumen

⇒ Frage von Lilyanne ~~von~~ (Tweed Test 2, zum entwerfen)

$h=6$

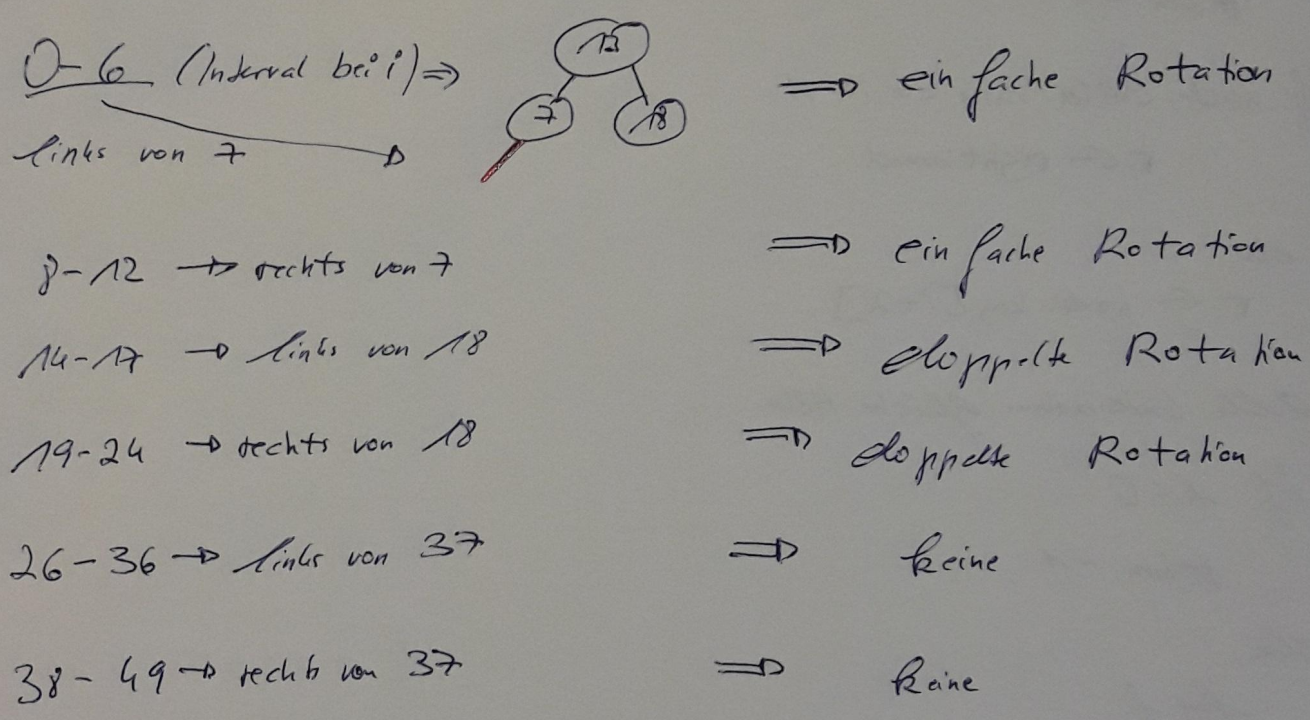
337 = ist kein Blatt

753 = befindet sich auf Ebene 3

Suchbaum eigenschaft erfüllt? ⇒ Ja, alles was links steht ist kleiner als die Wurzel, alles was rechts steht größer & auch innerhalb der Unterbäume ist das der Fall

Test 2 → 2b)

AVL Baum = balanciert (garantiert)
hat logarithmische Komplexität



} (Rest selbstständig lösen)

Anmerkung = das SORTIEREN mit heap ⇒ nicht relevant / nicht durchsprechen

B* Bäume → egal

A2) Bäume a) (Test 2) (Pseudocode wurde im Rep. nicht zur Baum beschrieben)

check Btree (node, leftbound, rightbound)

if node.child.size() == 0
return 1 // Definitionssache

else

h ← 0, a ← 0, l ← leftbound, r ← node.key[1]

for i ← 0 bis node.child.size() - 1

if (i == 0)

h ← check Btree (node.child[i], l, r)

else

a ← check Btree (node.child[i], l, r)

if a ≠ h

return -1

if node.child.size == i + 1

r ← rightbound

else

r ← node.key[i + 2]

Alle Unterbäume gleiche Höhe...

if h < 0

return -1

else

h + 1

- o ES fehlen noch Überprüfungen
- o Keys überprüfen leftbound & rightbound