

73. Geometrische Verteilung, Vorübung, 6 Punkte

- (a) (2 Punkte) Wie oft muss man (im Erwartungswert) mit einem einzelnen Würfel würfeln, bis man insgesamt drei Sechsen geworfen hat?
- (b) (4 Punkte) Gegeben sind n verschiedene Zahlen. Wir wählen zufällig eine Zahl a davon aus und bestimmen dann die Anzahl der n^- der Zahlen $< a$ und die Anzahl n^+ der Zahlen $> a$. Wir wiederholen diesen Versuch solange, bis $n^- \geq 10$ und $n^+ \geq 10$ ist. Was ist der Erwartungswert für die Anzahl der Versuche, in Abhängigkeit von n ?
- (c) (0 Punkte, schwierig) Wie oft muss man im Erwartungswert mit einem einzelnen Würfel würfeln, bis man drei Sechsen hintereinander geworfen hat?

74. Suche nach Rang, Textaufgabe, 7 Punkte

Erweitern Sie folgende Wörterbuch-Datenstrukturen (sofern das sinnvoll ist), so dass es effizient möglich ist, bei Eingabe von k den k -kleinsten Schlüssel in der gespeicherten Schlüsselmenge zu finden. Beschreiben und analysieren Sie auch die nötigen Modifikationen für die anderen Wörterbuchoperationen:

- ✓(a) Skipliste
- (b) Digitaler Suchbaum (*trie*, nicht komprimiert), mit lexikographischer Ordnung
- (c) Hashtabelle
- ✓(d) 2-3-Baum

75. Backtracking, 0 Punkte

Eine Halbdame kann sich wie eine Dame beim Schachspiel bewegen, außer dass sie nur in *einer* Diagonalenrichtung (von links unten nach rechts oben) ziehen kann und nicht in der anderen Diagonalenrichtung. (Sie kann nach wie vor waagrecht oder senkrecht ziehen.)

- (a) Wie viele Möglichkeiten gibt es, 8 Halbdamen auf das Schachbrett zu stellen, sodass sie sich nicht gegenseitig angreifen können?
- (b) Wie viele verschiedene Möglichkeiten gibt es, wenn man gedrehte oder gespiegelte Konfigurationen als gleich betrachtet?
- (c) Zeigen Sie, dass man auf jedem $(n \times n)$ -Schachbrett n Halbdamen unterbringen kann.

76. Drachenschach, 7 Punkte, Programmieraufgabe

Der Drache vereinigt die Bewegungsmöglichkeiten einer Dame und eines Springers im klassischen Schachspiel: Zusätzlich zu Zeilen, Spalten und Diagonalen kann ein Drache von Position (i, j) auf Position (i', j') ziehen, wenn $\{|i - i'|, |j - j'|\} = \{1, 2\}$ ist.

- (a) Wie viele Drachen, die sich nicht angreifen können, kann man auf dem 8×8 Schachbrett maximal aufstellen?
- (b) Was ist das kleinste $(n \times n)$ -Schachbrett mit $n \geq 2$, auf dem man n Drachen aufstellen kann?

Schreiben Sie ein Backtracking-Programm für diese Aufgaben. Zählen Sie die rekursiven Aufrufe insgesamt und auch getrennt für jede Ebene des Lösungsbaums.

Backtracking

Poly. Laufzeit $O(n^k)$
 exp. " " $O(c^n)$
 $A \leq_p B$

NP = Nondeterministische Poly..

NP ? P

def diag (xs, y, x):

for i in range (0, n):

if (xs[y][i] == True):

if (xs[i][x] == True):

x = 3

y = 4

y = 4 - 3 = 1

x = 0

y \ x	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								

x2 = 1
y2 = 0

x2 = 0
y2 = 1

y \ x	0	1
0		
1		

y \ x	0	1	2	3
0				
1				
2				
3				

y \ x	0	1	2
0			
1			
2			

def diag2 (xs, y, x):

x2 = x - 1

y2 = y - 1

while (x2 >= 0 and y2 >= 0):

if (xs[y2][x2] == True): break

x2 -= 1

y2 -= 1

x2 = x - 1

y2 = y + 1

while (x2 >= 0 and y2 <= 7):

if (xs[y2][x2] == True): break

x2 -= 1

y2 += 1

x2 = x + 1

y2 = y - 1

while (x2 <= 7 and y2 >= 0):

if (xs[y2][x2] == True): break

x2 += 1

y2 -= 1

x2 = x + 1

y2 = y + 1

while (x2 <= 7 and y2 <= 7):

if (xs[y2][x2] == True): break

x2 += 1

y2 += 1

- (a) (2 Punkte) Wie oft muss man (im Erwartungswert) mit einem einzelnen Würfel würfeln, bis man insgesamt drei Sechsen geworfen hat?
- (b) (4 Punkte) Gegeben sind n verschiedene Zahlen. Wir wählen zufällig eine Zahl a davon aus und bestimmen dann die Anzahl der n^- der Zahlen $< a$ und die Anzahl n^+ der Zahlen $> a$. Wir wiederholen diesen Versuch solange, bis $n^- \geq 10$ und $n^+ \geq 10$ ist. Was ist der Erwartungswert für die Anzahl der Versuche, in Abhängigkeit von n ?

a) $f = \text{eine Sechse}$ $E = 6$

$f = \text{drei Sechsen da unabhängig}$ $E = 3 \cdot 6$

$$3 \cdot \frac{1}{5} \cdot \left(\frac{5}{6} - 1\right)^2 = 3 \cdot 6 = 18$$

$$b) \sum_{k=1}^{\infty} k \cdot \frac{20^{k-1}}{24^k} = a_0 \cdot \frac{q}{(q-1)^2}$$

$$\Rightarrow \frac{20-n}{(n-1)} \cdot \frac{\frac{20}{n}}{\left(\frac{20}{n} - 1\right)^2} \neq \frac{n}{n-20}$$

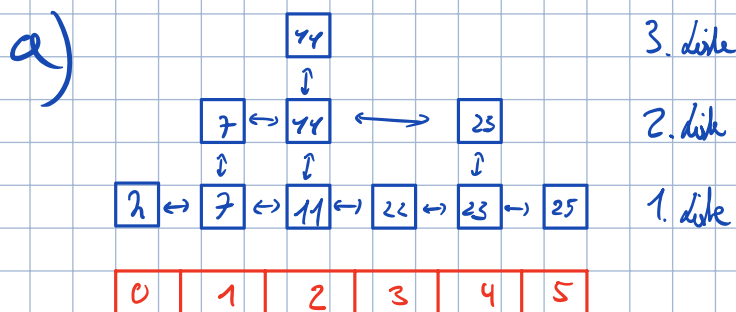
$$f \quad n=21 \quad E(x)=21$$

$$n=22 \quad E(x) \approx 10$$

74. Suche nach Rang, Textaufgabe, 7 Punkte

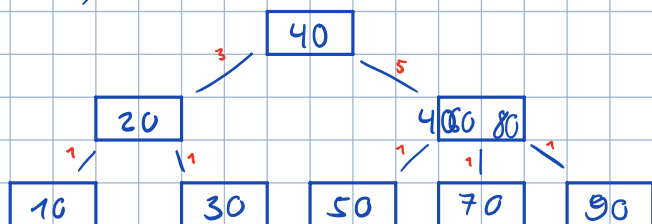
Erweitern Sie folgende Wörterbuch-Datenstrukturen (sofern das sinnvoll ist), so dass es effizient möglich ist, bei Eingabe von k den k -kleinsten Schlüssel in der gespeicherten Schlüsselmenge zu finden. Beschreiben und analysieren Sie auch die nötigen Modifikationen für die anderen Wörterbuchoperationen:

- ✓ (a) Skipliste
- (b) Digitaler Suchbaum (*trie*, nicht komprimiert), mit lexikographischer Ordnung
- (c) Hashtabelle
- ✓ (d) 2-3-Baum



Erweiterung: Merke den Index des 1. Liste und vergleiche diese, anstatt dem Inhalt der Liste, um den k -kleinsten Schlüssel zu finden. (Findet man, der bereits Sucht)

d) 2-3-Baum



Erweiterung: Merke dir in jedem Knoten die Anzahl Kinder, leftchild, middlechild, rightchild

Um den k -kleinsten Key zu finden, gehe nach folgender Bed. den Weg:

leftchild \geq key: gehe in diese Richtung

middlechild \geq key - leftchild: " — "

rightchild \geq (key - leftchild) - middlechild: " — "

c)

0	1	2	3	4	5	6	7	8
	28	20	12		5	15		17
	19					33		
	10							

mod 3 nicht möglich es möglichst effizient zu erweitern

b) root



Erweiterung: Nummerieren Angefangen vom Ende eines Knoten bis zum Root.

So kann das k -kleinste Element gefunden werden, indem man im Root nach dem k -kleinsten Wert sucht und dann den darauf folgenden Zahlen nach "unten" folgt.

(auch nicht gerade speicher effizient)

76. Drachenschach, 7 Punkte, Programmieraufgabe

Der Drachen vereinigt die Bewegungsmöglichkeiten einer Dame und eines Springers im klassischen Schachspiel: Zusätzlich zu Zeilen, Spalten und Diagonalen kann ein Drachen von Position (i, j) auf Position (i', j') ziehen, wenn $\{|i - i'|, |j - j'|\} = \{1, 2\}$ ist.

- (a) Wie viele Drachen, die sich nicht angreifen können, kann man auf dem 8×8 Schachbrett maximal aufstellen?
- (b) Was ist das kleinste $(n \times n)$ -Schachbrett mit $n \geq 2$, auf dem man n Drachen aufstellen kann?

a) Code

b) 10

```
C:\Users\Thore\Downloads\home\mi\thob97\alp3\u11>python main.py
Schachbrett= 1 x 1 , Rekursion= 1 , Anzahl= 1
Dauer: 0.0 s
Schachbrett= 2 x 2 , Rekursion= 5 , Anzahl= 1
Dauer: 0.0 s
Schachbrett= 3 x 3 , Rekursion= 14 , Anzahl= 1
Dauer: 0.0 s
Schachbrett= 4 x 4 , Rekursion= 70 , Anzahl= 2
Dauer: 0.0 s
Schachbrett= 5 x 5 , Rekursion= 615 , Anzahl= 4
Dauer: 0.03 s
Schachbrett= 6 x 6 , Rekursion= 6403 , Anzahl= 4
Dauer: 0.39 s
Schachbrett= 7 x 7 , Rekursion= 91324 , Anzahl= 5
Dauer: 6.89 s
Schachbrett= 8 x 8 , Rekursion= 1849660 , Anzahl= 6
Dauer: 173.73 s
```

Rekursion = Anzahl der Aufrufe der Funktion

Anzahl = Maximale mögliche Drachen