

## informatiCup 2013 • Aufgabe 3

### Würfelturm-Editor

#### Einführung

Noch in den 1970er Jahren wurde Computerbenutzern Rechenleistung und Speicherkapazität in Form von Großrechnern (engl. *mainframe*) bereitgestellt. Großrechner bedienten eine große Anzahl von Benutzern gleichzeitig. Der Zugriff erfolgte über Terminals. In den 1980er Jahren wurden die Terminals in vielen Bereichen von Einzelplatzrechnern (engl. *personal computer*) ersetzt, auf denen Anwendungen lokal ausgeführt und von einzelnen Benutzern bedient wurden. Seit den 1990er Jahren erlaubt das Internet den Online-Datenaustausch zwischen einzelnen Rechnern. Der Zugang zum Internet konnte mit immer höheren Datenübertragungsraten erfolgen und ermöglicht heute – wie schon in den 1970er Jahren – eine zentrale Bereitstellung von Rechenleistung und Speicherkapazität.

Anwendungen werden wieder in Rechenzentren ausgeführt und über das Internet benutzt. Solche Anwendungen können nicht nur von einem einzelnen Benutzer sondern von vielen Anwendern gleichzeitig verwendet werden. Zum Beispiel können mehrere Benutzer gleichzeitig ein gemeinsames Dokument bearbeiten.

In dieser Aufgabe soll die Echtzeit-Zusammenarbeit (engl. *live collaboration*) von Benutzern an einem gemeinsamen Dokument simuliert werden. Dieses Dokument beschreibt ein Bauwerk aus Würfeln. Die Benutzer sind die Baumeister dieses Würfelbauwerks.

#### 1 Aufgabe

In dieser Aufgabe werden die Probleme modelliert, eindimensionale Würfelgrundmauern, zweidimensionale Würfelwände und schließlich dreidimensionale Türme aus unterschiedlich gefärbten Würfeln von mehreren Baumeistern (d.h. Benutzern) gleichzeitig bearbeiten zu lassen. Konflikte, die entstehen können, wenn gleichzeitig an einem Bauwerk gearbeitet wird, sollen automatisch aufgelöst werden.

Die Würfelbauwerke bestehen aus verschiedenen jeweils einfarbigen Würfeln, deren Farbwerte in der RGB-Kodierung gegeben sind.

Die folgenden drei Operationen können auf die Würfelbauwerke angewendet werden.

- Einfügen eines Würfels mit der Farbe  $c$  an der Position  $pos$  durch Baumeister  $b$ :  $ins(pos,c,b)$
- Entfernen des Würfels an der Position  $pos$  durch Baumeister  $b$ :  $del(pos,b)$
- Einfärben des Würfels an der Position  $pos$  mit der Farbe  $c$  durch Baumeister  $b$ :  $sub(pos,c,b)$

Eine Farbe  $c$  referenziert ein RGB-Tripel, in dem jede Farbkomponente mit einem Byte kodiert ist.

Der Name  $b$  identifiziert einen Baumeister wie in Abschnitt 2 beschrieben.

Eine Position  $pos$  ist, abhängig davon ob das zu bearbeitende Bauwerk eine Würfelgrundmauer, eine Würfelwand oder ein Würfelturm ist, ein ein-, zwei- oder dreidimensionaler Vektor.

Ein Würfelbauwerk kann von beliebig vielen Baumeistern gleichzeitig bearbeitet werden (d.h. jeder Baumeister bearbeitet eine Kopie des aktuellen Bauwerks, die im Falle von Änderungen mit den Kopien der anderen Baumeister synchronisiert werden muss).

## 1.1 Ein Beispiel

Abb. 1 zeigt als Ergebnis von z.B. drei Einfügeoperationen eine eindimensionale Würfelgrundmauer bestehend aus einem roten (Position 0), grünen (Position 1) und blauen (Position 2) Würfel.

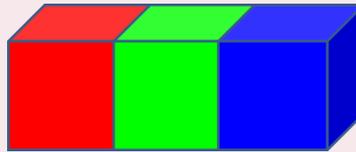
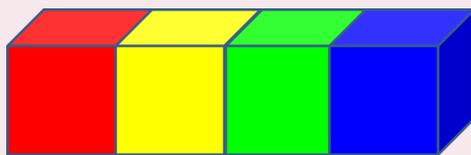
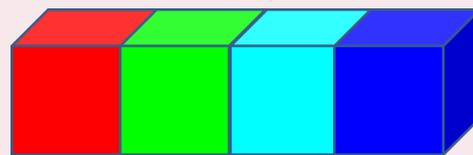


Abbildung 1: Eine Würfelgrundmauer mit drei Würfeln

Dieses Bauwerk wird nun von zwei Baumeistern gleichzeitig geändert. Baumeister A fügt an Position 1 (d.h. zwischen dem roten und dem grünen Würfel) einen gelben Würfel ein, Baumeister B fügt an Position 2 (d.h. zwischen dem grünen und dem blauen Würfel) einen türkisen Würfel ein. Die beiden Operationen sind also  $o_1 = \text{ins}([255,255,0],1,A)$  und  $o_2 = \text{ins}([0,255,255],2,B)$ . Entsprechend sieht die Würfelgrundmauer für die beiden Baumeister wie in Abb. 2 dargestellt aus.



(a) Baumeister A

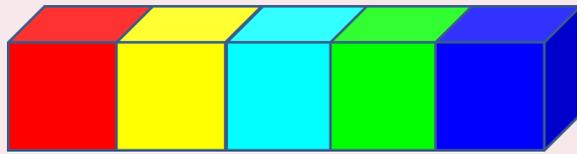


(b) Baumeister B

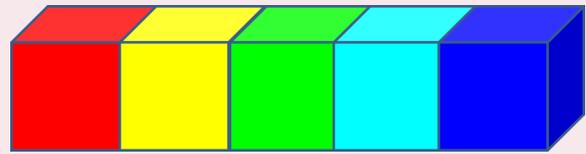
Abbildung 2: Die Änderungen der beiden Baumeister

Würden nun die beiden Operationen  $o_1$  und  $o_2$  (in dieser Reihenfolge) an die anderen Baumeister, Operation  $o_2$  an Baumeister A und Operation  $o_1$  an Baumeister B verteilt, so erhielten diese jeweils die in Abb. 3 dargestellten inkonsistenten Bauwerke.

Tatsächlich sieht in diesem Beispiel eine vereinheitlichte konsistente Würfelgrundmauer wie in Abb. 4 dargestellt aus: Der gelbe Würfel wird zwischen dem roten und dem grünen Würfel und der türkise Würfel wird zwischen dem grünen und dem blauen Würfel eingefügt.



(a) Baumeister A und alle anderen



(b) Baumeister B

Abbildung 3: Zwei inkonsistente Würfelgrundmauern

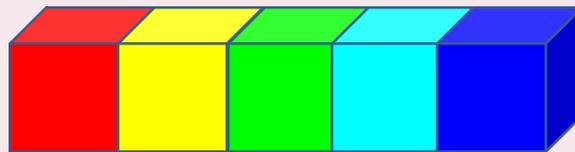


Abbildung 4: Die konsistente Würfelgrundmauer mit beiden Änderungen

## 1.2 Konsistente Würfelbauwerke

Ein synchronisiertes Würfelbauwerk ist *konsistent*, wenn die von den einzelnen Baumeistern beabsichtigten Farbverläufe beibehalten werden – unabhängig von den absoluten Würfelpositionen im synchronisierten Bauwerk. In dem Beispiel in Abschnitt 1.1 ist daher wichtig, dass der gelbe Würfel von Baumeister A zwischen dem roten und dem grünen Würfel eingefügt wird. Genauso muss der türkise Würfel von Baumeister B auch in dem synchronisierten Bauwerk zwischen dem grünen und dem blauen Würfel erscheinen.

## 1.3 Semantik der Einfüge-, Lösch- und Ersetzungsoperationen

Für alle Arten von Würfelbauwerken sind Einfügeoperationen so durchzuführen, dass der gegebene Würfel an der gegebenen Position im Bauwerk (d.h. der lokalen Kopie des Baumeisters) eingefügt wird. Befindet sich an dieser Stelle bereits ein Würfel, so werden die  $x$ -Koordinaten dieses und aller auf der  $x$ -Achse bis zur ersten unbesetzten Position folgenden Würfel um eins erhöht (siehe Abb. 5).

Einfügeoperationen sind grundsätzlich an allen Positionen möglich – auch dann wenn der eingefügte Würfel mit keinem anderen Würfel in Zusammenhang steht.

Für alle Arten von Würfelbauwerken sind Löschoptionen so durchzuführen, dass der Würfel an der gegebenen Position im Bauwerk (d.h. der lokalen Kopie des Baumeisters) gelöscht wird. Die  $x$ -Koordinaten aller auf der  $x$ -Achse bis zur ersten unbesetzten Position folgenden Würfel werden um eins erniedrigt.

Löschoptionen an Positionen, an denen sich kein Würfel befindet, haben keinen Effekt.

Ersetzungen an Positionen, an denen sich kein Würfel befindet, fügen den gegebenen Würfel an der gegebenen Position im Bauwerk (d.h. der lokalen Kopie des Baumeisters) ein.

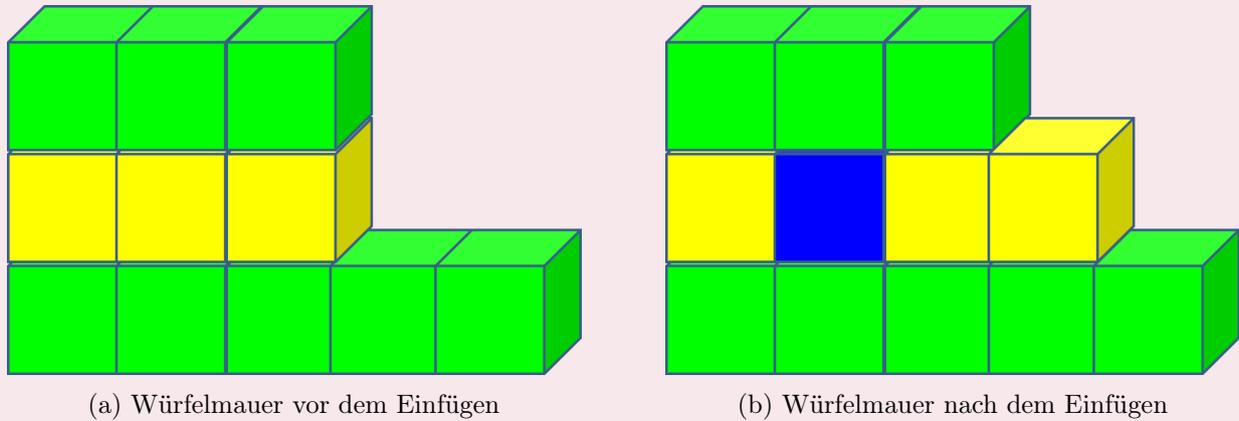


Abbildung 5: Einfügen eines Würfels in eine Würfelmauer

## 1.4 Erste Runde

Berechnen Sie für eine gegebene Folge von Operationen auf Würfelbauwerken (d.h. eindimensionale Würfelgrundmauern, zweidimensionale Würfelwände und dreidimensionale Würfeltürme) das resultierende konsistente Würfelbauwerk. Geben Sie dieses Würfelbauwerk im unten beschriebenen Format aus. Für den Fall, dass keine automatische Konfliktauflösung möglich ist (z.B. bei zwei gleichzeitigen Ersetzungen der Farbe eines Würfels), brechen Sie die Berechnung ab und geben das aktuelle Würfelbauwerk und die nicht zu vereinheitlichenden Operationen aus.

Hinweis: Ein Baumeister kann auf seiner lokalen Kopie eines Würfelbauwerks durchaus mehrere Operationen durchführen, bevor synchronisiert wird.

Erstellen Sie mindestens jeweils drei Testfälle für Würfelgrundmauern, Würfelmauern und Würfeltürme (im unten beschriebenen Format), von denen Sie annehmen, dass Ihr Programm im Vergleich zu anderen Programmen gute Ergebnisse liefert.

## 1.5 Zweite Runde

Erstellen Sie eine graphische Benutzeroberfläche, die ein gegebenes Würfelbauwerk und den Ablauf ihres Synchronisationsalgorithmus' mit den Positionen und Farben der Würfel in dem Bauwerk visualisiert.

Nicht alle Konfliktsituationen lassen sich automatisch auflösen. Vielleicht bietet Ihre Benutzeroberfläche daher zusätzlich die Möglichkeit, manuelle Änderungen an dem entstehenden Würfelbauwerk vorzunehmen, um eine Synchronisation zu erreichen.

## 2 Eingabeformat

Ihr Programm verarbeitet “Plain text”-Dateien. Die erste Zeile enthält die Anzahl  $k$  der verfügbaren Farben. Die folgenden  $k$  Zeilen enthalten die Farben  $c_1..c_k$ . Die Farben sind als RGB-Werte im Format  $r\ g\ b$  gegeben. Als Eingabewerte sind für  $r$ ,  $g$  und  $b$  ganze Zahlen von 0 bis 255 erlaubt. Es folgt die Anzahl der Dimensionen  $d \in \{1, 2, 3\}$ . Die folgenden Zeilen enthalten die Operationen auf einem initial leeren Würfelbauwerk.

Die Operationen für das Einfügen, Löschen und Ersetzen von Würfeln entsprechen den folgenden regulären Ausdrücken.

- Einfügen:

```
ins\(\(\d+(\,\d+)?(\,\d+)?\),\d+,\w+\)
```

mit der ein-, zwei- oder dreidimensionalen Position des Würfels, gefolgt von der Referenz einer der Farben  $c_1..c_k$  und dem Namen des Baumeisters.

Beispiel: Die Operation `ins((4,2,3),0,Alfred)` fügt für Baumeister Alfred einen Würfel mit der nullten Farbe in der Farbenliste an der Position (4,2,3) in einen Würfelturm ein.

- Löschen:

```
del\(\(\d+(\,\d+)?(\,\d+)?\),\w+\)
```

mit der ein-, zwei- oder dreidimensionalen Position des Würfels und dem Namen des Baumeisters.

Beispiel: Die Operation `del((4,2),Bruno)` löscht für Baumeister Bruno den Würfel an der Position (4,2) in einer Würfelmauer.

- Einfärben/Ersetzen:

```
sub\(\(\d+(\,\d+)?(\,\d+)?\),\d+,\w+\)
```

mit der ein-, zwei- oder dreidimensionalen Position des Würfels, gefolgt von der Referenz einer der Farben  $c_1..c_k$  und dem Namen des Baumeisters.

Beispiel: Mit der Operation `sub((4),1,Clara)` färbt Baumeisterin Clara den Würfel an der vierten Position in einer Würfelgrundmauer mit der ersten Farbe in der Farbenliste ein.

Die Nummerierung der Farben und Positionen ist null-basiert.

Die Synchronisation erfolgt schrittweise pro Zeile.

Eine gültige Eingabe kann in einer Zeile mehrere Operationen pro Baumeister enthalten. Stehen in einer Zeile mehrere Operationen der gleichen Baumeister, so sind diese für jeden Baumeister in der entsprechenden Reihenfolge anzuwenden (zwischen den Operationen eines Baumeisters können dabei in dieser Zeile Operationen anderer Baumeister stehen).

Die folgende Eingabe enthält das Beispiel aus Abschnitt 1.1.

```
5
255 0 0
0 255 0
0 0 255
255 255 0
0 255 255
1
ins((0),0,Start)
ins((1),1,Start)
ins((2),2,Start)
ins((1),3,A) ins((2),4,B)
```

### 3 Ausgabeformat

Ihr Programm gibt “Plain text”-Dateien aus. Die erste Zeile enthält die Anzahl  $k$  der in dem Würfelbauwerk verwendeten Farben. Die folgenden  $k$  Zeilen enthalten die Farben  $c_1..c_k$ . Die Kodierung der Farben entspricht dem Eingabeformat. Die folgende Zeile enthält die Anzahl  $n$  der in einem Würfelbauwerk nach Anwendung aller synchronisierten Operationen enthaltenen Würfel. Die nächste Zeile enthält die Anzahl der Dimensionen  $d \in \{1, 2, 3\}$  des Würfelbauwerks. Die folgenden  $n$  Zeilen enthalten jeweils die Information über Position und Farbe eines Würfels, entsprechend dem folgenden regulären Ausdruck.

- Position und Farbe eines Würfels:

$$\backslash(\backslash d+(, \backslash d+)(, \backslash d+)?\backslash) \backslash d+$$

mit der ein-, zwei- oder dreidimensionalen Position des Würfels, gefolgt von einer Referenz auf eine der Farben  $c_1..c_k$ .

Beispiel: Ein Würfel mit der fünften Farbe in der Farbenliste an der Position (4,2,3) in einem Würfelturm wird mit dem Eintrag (4,2,3) 5 beschrieben.

Für den Fall, dass sich die Operationen in einer Zeile der Eingabedatei nicht synchronisieren lassen, so geben sie abschließend diese Zeile in der Ausgabedatei aus.

Die folgende Ausgabe beschreibt die synchronisierte Würfelgrundmauer aus Abschnitt 1.1.

```
5
255 0 0
0 255 0
0 0 255
255 255 0
0 255 255
5
1
(0) 0
(1) 3
(2) 1
(3) 4
(4) 2
```

#### 4 Außerdem...

Erstellen Sie für Ihre Implementierung bitte eine Bedienungs- und Installationsanleitung. Dokumentieren Sie die von Ihnen getroffenen Entscheidungen bei der Auswahl verwendeter Algorithmen und Datenstrukturen und in der Software-Entwicklung.