

# Mathe Dual Programmierwettbewerb 2022

## Holzspielzeug für die Matse Spaß AG

Die Matse Spaß AG steht für die Erfindung von Holzspielzeugen und Holzpuzzeln. Für die neuste Erfindung braucht die Firma jedoch ein Programm zur Überprüfung, ob ein Holzpuzzle sich lösen lässt oder nicht.

Daher möchte die Matse Spaß AG von Ihnen für ihr neu entwickeltes Holzspielzeug ein Programm entwickeln lassen, welches Holzbauteile über eine Eingabedatei erfasst und eine mögliche Lösung beziehungsweise Anordnung der Puzzleteile ausgibt.

Die einzelnen Puzzleteile sind dabei Holzwürfel an deren Seiten Dreiecke sind. Die Aufgabe besteht nun darin, die Würfel so zusammenzusetzen, dass die Dreiecke zweier benachbarter Würfel wieder ein Quadrat bilden. Durch die unterschiedliche Anordnung der Dreiecke auf dem Würfel gibt es eine Vielzahl von Kombinationen. Da sich die kleinen Würfel auch beliebig drehen lassen, gibt es auch Kombinationen die nicht sinnvoll sind, da dann Dreiecke nach Außen zeigen oder nicht aufeinanderpassen.

Leider hat die Firma keinerlei Ahnung wie man sinnvoll die einzelnen Bauteile möglichst einfach digital erfasst oder eingibt.

Sie erstellt jedoch für den Schreiner Zeichnungen der die einzelnen Würfel zurechtschneidet und die Dreiecke an den Würfel klebt.

Die Zeichnungen sind wie an dem rechten Beispiel dargestellt. Die blauen Dreiecke sind dann die Flächen, auf denen die zusätzlichen Dreiecke geklebt werden. Ein Beispiel mit vier kleinen Würfeln ist in Abbildung 2 zu sehen.

Abbildung 2: Teile eines Würfels mit angesetzten Dreiecken

Eingabe und Ausgabe definiert.

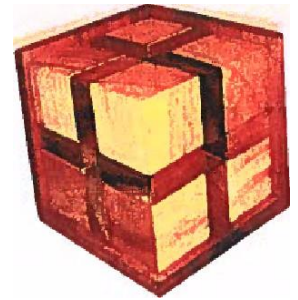


Abbildung 1: zusammengesetzter 8-teiliger Würfel

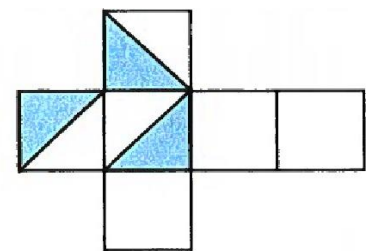






Abbildung 3: Zeichnung eines kleinen Würfels als Risszeichnung

## Definition der Flächen

Betrachtet man eine Seite eines Würfels, so gibt es die nachfolgenden Möglichkeiten von Flächen. Jede Seite kann ein Dreieck enthalten oder leer sein. Die mögliche Zuordnung der Dreiecke wird als eine Ziffer angegeben.

| Seite mit oder ohne Dreieck |  |  |  |  |  |
|-----------------------------|---|---|---|---|---|
| Ziffer                      | 0   | 1   | 2   | 3   | 4   |

Die blauen Dreiecke repräsentieren jeweils ein anliegendes Dreieck.

## Eingabe der kleinen Würfel

|   |   |   |   |
|---|---|---|---|
|   | 1 |   |   |
| 2 | 3 | 4 | 5 |
|   | 6 |   |   |

Abbildung 4: Risszeichnung des Würfels mit Zuordnung der Flächen

Die Lage der sechs Flächen eines kleinen Würfels als Risszeichnung ist so, dass die Fläche 1 nach oben und die Fläche 3 immer nach vorne zeigt. Dementsprechend zeigt die Fläche 2 nach links, die Fläche 4 nach rechts und die Fläche 3 nach hinten.

Für den Würfel aus Abbildung 3 ergeben sich dadurch die Werte 1 2 4 0 0 0 jeweils durch ein Leerzeichen getrennt.

Jeder Würfel hat eine zusätzliche Bezeichnung, z. B. Teil 1 vor den Werten. Zur Trennung der Bezeichnung von den nachfolgenden Werten wird ein Doppelpunkt verwendet.

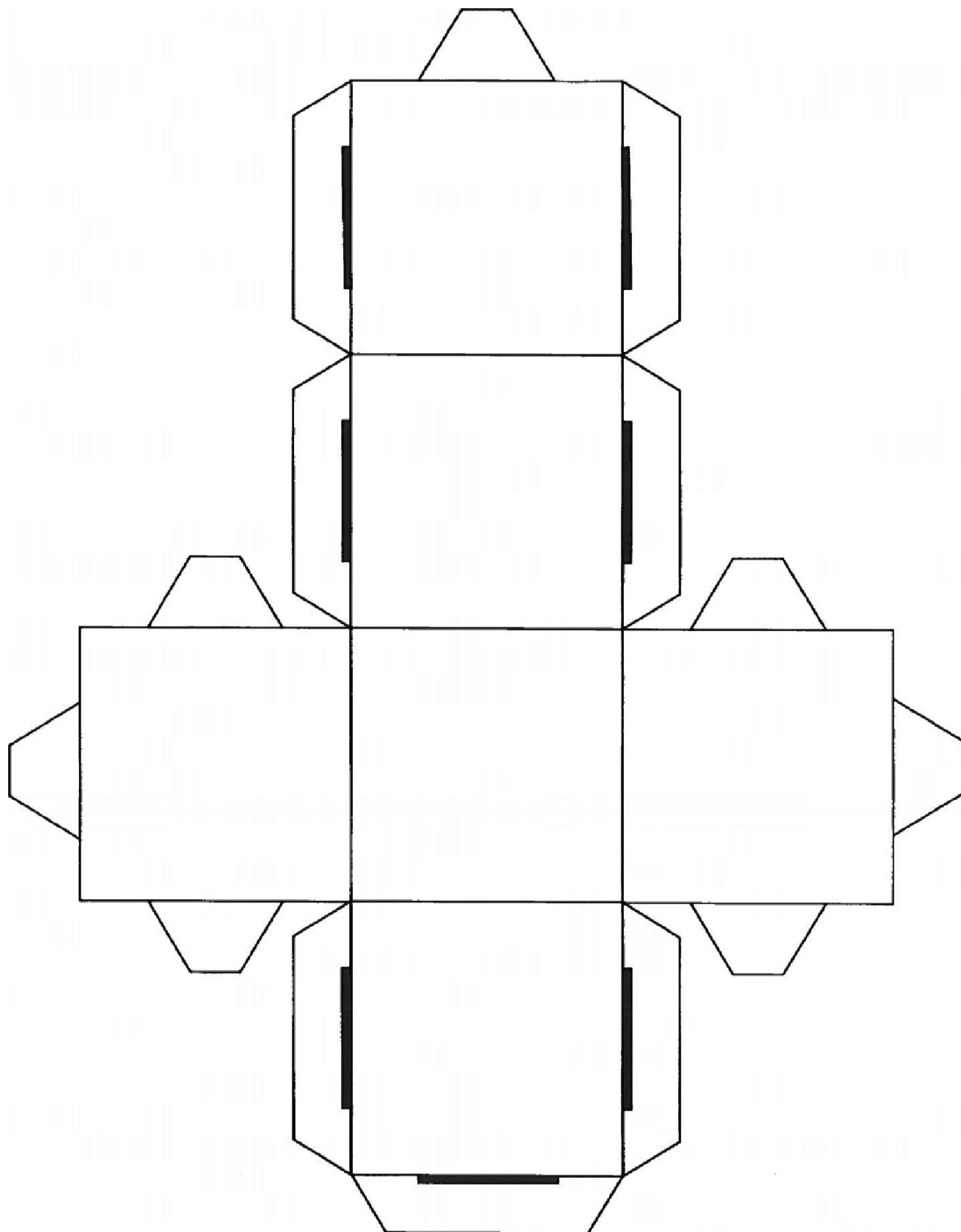
Dadurch ergibt sich für die Eingabe des kleinen Würfels aus Abbildung 3

Teil 1: 1 2 4 0 0 0

Dieses Blatt kann an der Perforation aus dem Aufgabensatz herausgetrennt werden!

### Vorlage Würfel zum Ausschneiden und Falten.

Die dicken Linien sollen Schlitze in denen die laschen eingesteckt werden können.



## Dimension des Puzzles

Die Matse Spaß AG möchte einen Algorithmus, der alle möglichen Holzpuzzle lösen kann, Dazu muss die Dimension des Puzzles bekannt sein. im ersten Schritt soll die Form des Puzzles auf quaderförmige Puzzle beschränkt werden. Andere Formen sind schon angedacht, spielen für die Aufgabe aber keine Rolle.

In der Abbildung 1 ist ein Puzzle der Dimension  $2 \times 2 \times 2$  zu sehen. In den nachfolgenden Beispielen sind noch Puzzle der Größe  $3 \times 3 \times 2$  und  $1 \times 2 \times 3$  zu sehen. Jedes Puzzle lässt sich durch das Umlegen (Rotation) des Puzzles in ein anderes Puzzle umformen. Z.B. lässt sich das  $1 \times 2 \times 3$ -Puzzle durch Rotation in ein  $2 \times 3 \times 1$ -Puzzle wandeln. Die Musterlösung ist dann entsprechend anders.

Die Dimension eines Puzzles wird am Anfang der Eingabedatei angegeben und gibt dann auch vor, wie viele kleine Würfel eingelesen werden müssen.

Die Eingabe ist      Dimension  $\langle X \rangle, \langle Y \rangle, \langle Z \rangle$       also z.B. Dimension 1,2,3

## Drehachsen

Bei der Aufgabe sind die Drehungen der kleinen Würfel zu berücksichtigen. D.h. je nachdem wie man die Würfel dreht, gibt es auch andere mögliche Gesamtlösungen. Dreht man z.B. den Würfel aus Abbildung 3 um  $90^\circ$  um die Z-Achse so erhält man eine andere Ausgabe in der Musterlösung, welche aber identisch zu der anderen Lösung ist, bevor der Würfel gedreht wurde. Es gibt drei Achsen, die beim Finden der Lösung und der Rotation der kleinen Würfel zu berücksichtigen sind.

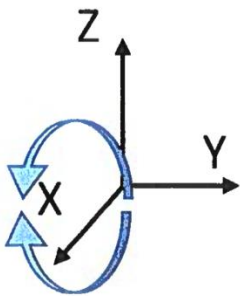


Abbildung 5: Drehung um X-Achse

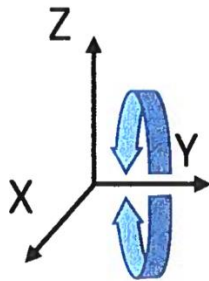


Abbildung 6: Drehung um Y-Achse

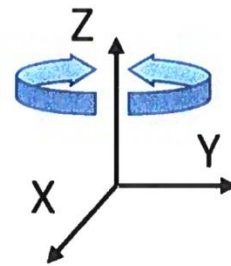


Abbildung 7: Drehung um Z-Achse

In dem folgenden Beispiel wird ein kleiner Würfel jeweils um  $90^\circ$  um die angegebene Achse gedreht. Dabei handelt es sich also immer um denselben Würfel! Der Würfel hat drei angesetzte Dreiecke, die hier der Übersicht halber unterschiedlich eingefärbt sind. Durch die Rotation ändert sich die Lage der Dreiecke sowie die Ausrichtung der Dreiecke. Zum Nachvollziehen befindet sich ein Pappwürfel zum Zusammenfalten auf der perforierten Anlage. Die nachfolgende Abbildung zeigt die Auswirkung der Rotation auf die Lage der Dreiecke auf einem kleinen Würfel.



Abbildung 8: kleiner Holzwürfel

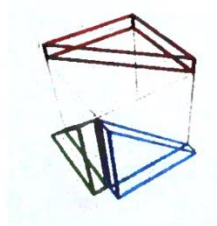


Abbildung 9: Würfel Risszeichnung

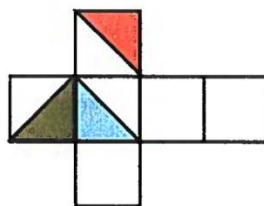


Abbildung 10: Risszeichnung Ausgangslage

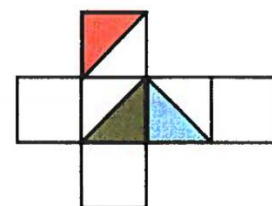


Abbildung 11:  $90^\circ$ -Drehung um Z-Achse

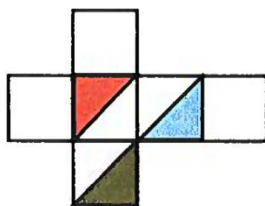


Abbildung 12: nach  $-90^\circ$ -Drehung um Y-Achse

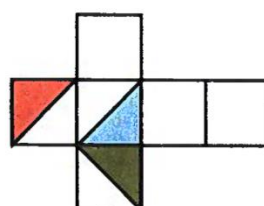


Abbildung 13: nach weiterer  $-90^\circ$ -Drehung um Z-Achse

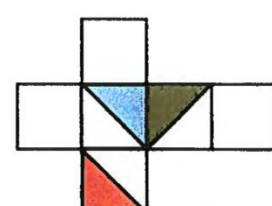


Abbildung 14: abschließend  $-90^\circ$ -Drehung um die X-Achse

Dreht man einen kleinen Würfel 4-mal um dieselbe Achse, erhält man wieder die Ausgangslage des Würfels.

## Eingabe

Die Eingabedatei hat das folgende Aussehen.

Jede Datei beginnt mit einem Header, in dem das Puzzle beschrieben ist. Dabei beginnen Kommentare immer mit einem doppelten Schrägstrich. Anschließend wird die Dimension angegeben und nachfolgend die Würfel, wie oben beschrieben, angehängt.

```
//*****  
// Beschreibung des Puzzles  
//*****  
Dimension <X>,<Y>,<Z>  
Teil 1: 1 2 4 0 0 0  
Teil 2: 2 2 3 0 0 0  
...  
Teil n: 1 1 1 0 0 0
```

## Ausgabe

Die Ausgabe erfolgt im ähnlichen Format wie die Eingabe. Die Würfel werden mit ihren Koordinaten und der Lage der einzelnen Dreiecke ausgegeben. Zuerst wird die Dimension des Puzzles angegeben. Anschließend werden die einzelnen Würfel mit ihrer Position, der Bezeichnung und der Ausrichtung angegeben. Die Angabe der Position erfolgt durch (x, y, z) wobei die Werte zwischen 1 und der jeweiligen Dimension liegen. Die Reihenfolge der Ausgabe der kleinen Würfel ist dabei egal! In den Beispielen werden diese in der Dimension absteigend angezeigt.

Für den Würfel in Abbildung 1 und dem Beispiel 3 ist die Musterlösung

```
//*****  
// Beschreibung des Puzzles  
//*****  
Dimension 2,2,2  
[2,2,2] Teil 3: 0 2 0 0 2 4  
[1,2,2] Teil 7: 0 2 1 0 0 3  
[2,1,2] Teil 5: 0 0 0 1 1 3  
[1,1,2] Teil 8: 0 0 2 1 0 2  
[2,2,1] Teil 6: 1 2 0 0 1 0  
[1,2,1] Teil 4: 2 1 2 0 0 0  
[2,1,1] Teil 2: 2 0 0 1 3 0  
[1,1,1] Teil 1: 3 0 4 2 0 0
```

Gibt man die Würfel in einer anderen Reihenfolge ein, ergibt sich zum gleichen Puzzle die folgende Musterlösung!

```
//*****  
// Beschreibung des Puzzles  
//*****  
Dimension 2,2,2  
[2,2,2] Teil 4: 0 4 0 0 2 1  
[1,2,2] Teil 5: 0 1 1 0 0 1  
[2,1,2] Teil 8: 0 0 0 3 1 1  
[1,1,2] Teil 6: 0 0 2 2 0 3  
[2,2,1] Teil 7: 4 2 0 0 4 0  
[1,2,1] Teil 3: 4 1 3 0 0 0  
[2,1,1] Teil 2: 4 0 0 1 1 0  
[1,1,1] Teil 1: 2 0 2 2 0 0
```

Das heißt für Ihre Lösung, dass die Ausgabe der Lösung sehr stark von Ihrem Algorithmus abhängt.

Es ist nicht gefordert, dass Sie die gleichen Ausgaben produzieren, wie sie in den Beispielen angegeben sind, da durch Rotation des Spiels auch weitere Lösungen möglich sind, Sie müssen jedoch die Dimension des Puzzles berücksichtigen und einhalten.

## Aufgabe

Entwerfen Sie für die Matse Spaß AG ein Programm, welches aus einer Datei die Konfiguration der einzelnen Würfel einliest und dafür eine Musterlösung ausgibt, sofern es eine Lösung gibt.

Lesen Sie aus einer Eingabedatei, die dem oben und nachfolgend beschriebenen Format entspricht, die kleinen Würfel sowie die Dimension des Puzzles ein.

Beachten Sie, dass man nicht nur Puzzle mit acht kleinen Würfeln ( $2 \times 2 \times 2$ ) sondern auch Puzzle mit verschiedenen Größen lösen kann, z.B. nur eine Ebene mit ( $1 \times 2 \times 2$ ) oder mit drei Ebenen also mit 27 kleinen Würfeln ( $3 \times 3 \times 3$ ). Es ist nicht gefordert für jede mögliche Dimension ein Beispiel anzugeben. Das Finden einer Lösung eines ( $2 \times 3 \times 3$ ) Puzzles und größer ist sehr zeitaufwendig!

Schreiben Sie einen Algorithmus der für die Puzzle die Lösbarkeit prüft und eine Musterlösung erstellt.

Testen Sie die Puzzles jeweils auf ihre Lösbarkeit. In einigen Fällen gibt es Würfel mit 2, 3, 4, 5 oder 6 Dreiecken. Testen Sie Ihr Verfahren im Falle einer Lösung auch mit einer anderen Anordnung der Dimensionen. D.h. ein Puzzle mit vier Teilen und der Dimension  $1 \times 2 \times 2$  dann als  $2 \times 2 \times 1$  oder  $2 \times 1 \times 2$ . Gibt es für das erste Puzzle eine Lösung, muss es auch für die anderen Puzzles eine Lösung geben.

Erstellen Sie jeweils eine Musterlösung, wie oben angegeben, zu den Puzzles, in welcher die Dimension des Puzzles, die Position der Teile und die Lage der Dreiecke angegeben werden, d. h. wie die kleinen Würfel innerhalb des Puzzles liegen.

## Komplexität

Zum Beispiel hat ein Holzpuzzle aus vier Würfeln  $4! = 4 \cdot 3 \cdot 2 \cdot 1 = 24$  Möglichkeiten, die Würfel auf den vier Positionen zu platzieren. An jeder Position hat jeder Würfel sechs Seiten, die jeweils nach vorne zeigen können und jede nach vorne zeigende Seite kann von der Ausgangsposition dreimal gedreht werden, ohne dass die entsprechende Seite nicht mehr nach vorne zeigt. D.h. die Seite eines jeden Würfels hat also vier verschiedene mögliche Lagen. Dadurch gibt es  $4! \cdot (6 \cdot 4)^4 = 24 \cdot 24^4 = 24^5 = 7.962.624$  mögliche Kombinationen.

Da sich die Würfel nicht ändern, also immer sechs Seiten haben, ist im Allgemeinen die Anzahl aller sinnvollen und nicht sinnvollen Kombinationen  $k(n) = n! \cdot 24^n$ .

Da diese Anzahl der Kombinationen schon für kleine  $n$  auch von einem Rechner kaum noch zu bewältigen sind, sind hier gute Strategien gefragt. Überlegen Sie daher vorab, welche Kombinationen schon mit weniger als den  $n$  gegebenen Würfeln nicht mehr zu einer Lösung des Puzzles führen können. Beispielsweise passen schon die ersten beiden Würfel in vielen Fällen nicht direkt aneinander oder Dreiecke zeigen nach außen, Diese Fälle braucht man nicht weiter zu verfolgen!

## Optimierung

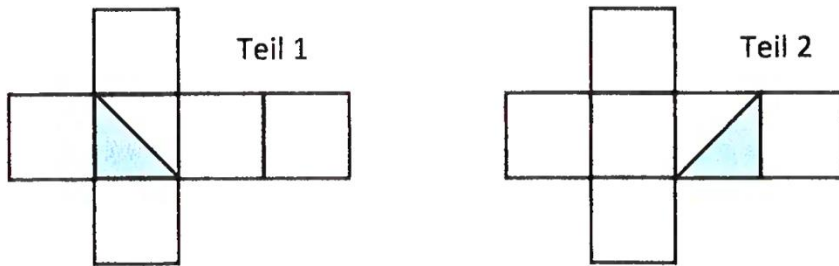
Für den Wettbewerb werden auch Testfälle generiert, die von einem Standard-Algorithmus nicht in der zugelassenen Zeit gelöst werden können. Nehmen Sie die Herausforderung an und überlegen sich Lösungsansätze mit denen Sie die Komplexität herunterbrechen (z.B. heuristische Verfahren für eine geeignete Vorauswahl von Lösungskandidaten) oder probieren Sie „parallele Programmierungsansätze“ aus. Da es eine ganze Reihe von Testfällen gibt, können auch Ansätze mit einer hohen Wahrscheinlichkeit eine richtige Lösung zu finden (ohne eine Garantie immer die korrekte Lösung zu liefern) eine gute Strategie sein. Im Verlaufe des Wettbewerbs haben Sie die Möglichkeit den aktuellen Stand der anderen Teilnehmer mit Ihrer eigenen Gesamt-Performance zu vergleichen und können dadurch Ihre Strategie dem aktuellen Stand anpassen.

## Testbeispiele

Für einen einfachen Vortest gibt es in Ihrem Git-Repository bereits einige einfache Beispiele. Diese werden durch das CI-System automatisch durchgetestet und dienen gleichzeitig der Qualifikation für den Performance-Test. Die ersten drei dieser Beispiele werden auf den folgenden Seiten vorgestellt und illustriert.

## Beispiel 1

Ein einfaches Beispiel, für welches die Lösung offensichtlich ist, besteht aus lediglich zwei kleinen Würfeln mit je einem angesetzten Dreieck.



Durch das Drehen der beiden Teile kann man diese schnell zusammensetzen. Die nachfolgenden Abbildungen zeigen zwei mögliche Lagen der beiden Würfel. Die zweite Lösung wird zusätzlich noch als Drahtmodell angezeigt.

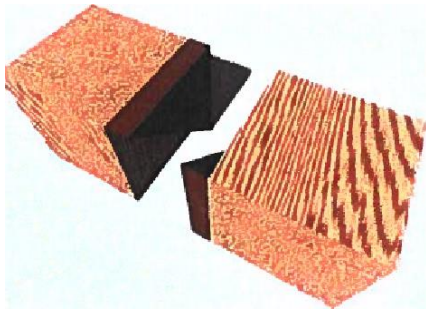


Abbildung 15: Lösung zu Beispiel 1

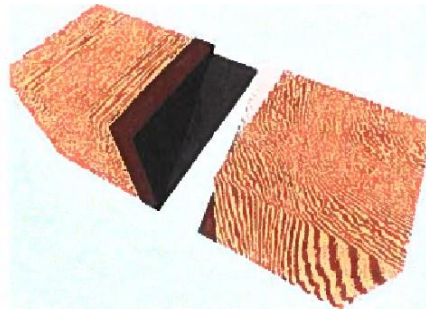


Abbildung 16: weitere Lösung

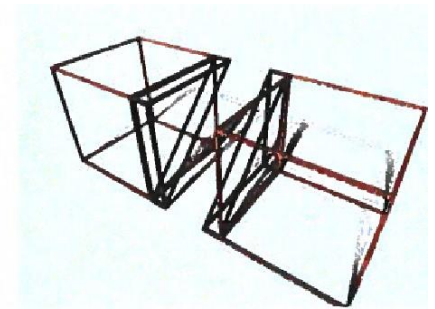


Abbildung 17: Drahtmodell zur Lösung

Die Eingabedatei

```
//*****  
// Beispiel 1  
//*****  
Dimension 1,2,1  
Teil 1: 0 0 1 0 0 0  
Teil 2: 0 0 0 4 0 0
```

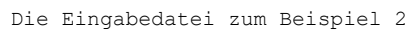
ergibt die nachfolgende mögliche Musterlösung.

```
//*****  
// Beispiel 1  
//*****  
Dimension 1,2,1  
[1,2,1] Teil 2: 0 4 0 0 0 0  
[1,1,1] Teil 1: 0 0 0 3 0 0
```

Für die Dimension 1,1,2 ergibt sich eine andere Lösung.

```
//*****  
// Beispiel 1  
//*****  
Dimension 1,1,2  
[1,1,2] Teil 2: 0 0 0 0 0 4  
[1,1,1] Teil 1: 1 0 0 0 0 0
```

Das nachfolgende Beispiel ist ein Bauplan für ein 6-teiliges (1 x 2 x 3) Puzzle.



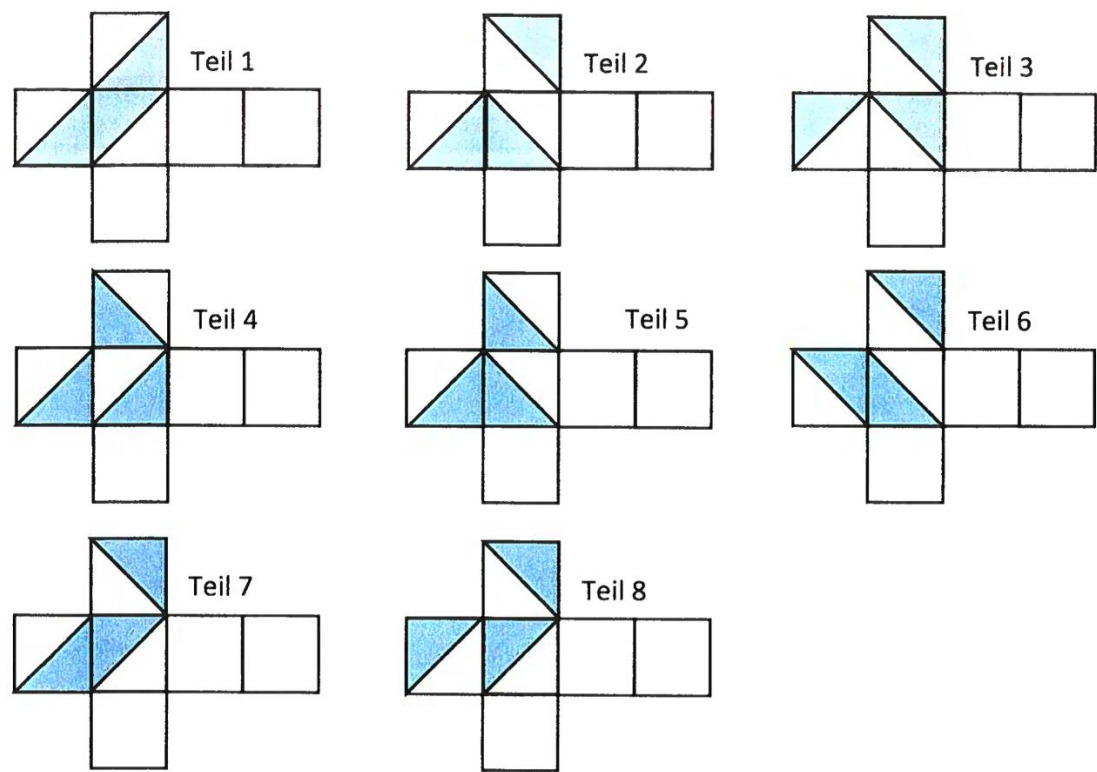
Mögliche Musterlösung ist

```
//*****
// Beispiel 2
//*****
Dimension 2,3,1
[2,3,1] Teil 6: 0 4 0 0 1 0
[1,3,1] Teil 2: 0 1 2 0 0 0
[2,2,1] Teil 3: 0 1 0 3 4 0
[1,2,1] Teil 4: 0 4 3 2 0 0
[2,1,1] Teil 5: 0 0 0 2 4 0
[1,1,1] Teil 1: 0 0 3 3 0 0
```



Beispiel 3

Das nachfolgende Beispiel ist ein Bauplan für einen 8-teiligen Würfel, bei dem die einzelnen kleinen Würfel zusammengesetzt den Würfel wie in der Abbildung 1 ergeben.



Die Eingabedatei zum Beispiel 3

```
//*****  
// Beispiel 3  
//*****  
Dimension 2,2,2  
Teil 1: 4 4 2 0 0 0  
Teil 2: 3 4 1 0 0 0  
Teil 3: 3 2 3 0 0 0  
Teil 4: 1 4 4 0 0 0  
Teil 5: 1 4 1 0 0 0  
Teil 6: 3 3 1 0 0 0  
Teil 7: 3 4 2 0 0 0  
Teil 8: 3 2 2 0 0 0
```

Mögliches Ergebnis

```
//*****  
// Beispiel 3  
//*****  
Dimension 2,2,2  
[2,2,2] Teil 4: 0 4 0 0 2 1  
[1,2,2] Teil 5: 0 1 1 0 0 1  
[2,1,2] Teil 8: 0 0 0 3 1 1  
[1,1,2] Teil 6: 0 0 2 2 0 3  
[2,2,1] Teil 7: 4 2 0 0 4 0  
[1,2,1] Teil 3: 4 1 3 0 0 0  
[2,1,1] Teil 2: 4 0 0 1 1 0  
[1,1,1] Teil 1: 2 0 2 2 0 0
```