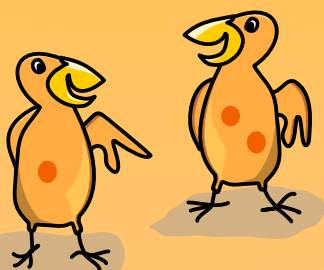




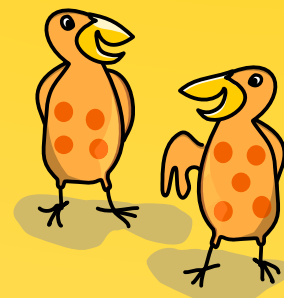
**INFORMATIK-BIBER SCHWEIZ**  
**CASTOR INFORMATIQUE SUISSE**  
**CASTORO INFORMATICO SVIZZERA**

## Aufgaben und Lösungen 2021

### Schuljahre 3/4



<https://www.informatik-biber.ch/>



Herausgeber:

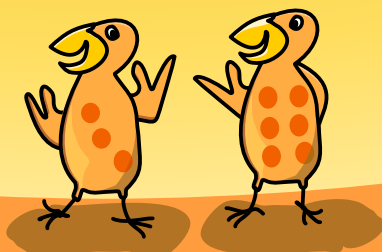
Susanne Datzko, Fabian Frei,  
Jean-Philippe Pellet



010100110101011001001001  
010000010010110101010011  
010100110100100101000101  
001011010101001101010011  
010010010100100100100001

# SV!A

[www.svia-ssie-ssii.ch](http://www.svia-ssie-ssii.ch)  
schweizerischerverein für informatik in d  
erausbildung // société suisse pour l'infor  
matique dans l'enseignement // società sviz  
zera per l'informatica nell'insegnamento







# Mitarbeit Informatik-Biber 2021

Masiar Babazadeh, Susanne Datzko, Fabian Frei, Martin Guggisberg, Gabriel Parriaux, Jean-Philippe Pellet

Projektleitung: Nora A. Escherle

Herzlichen Dank für die Aufgabenentwicklung für den Schweizer-Wettbewerb an:

Juraj Hromkovič, Michael Barot, Christian Datzko, Jens Gallenbacher, Dennis Komm, Regula Lacher, Peter Rossmanith: ETH Zürich, Ausbildungs- und Beratungszentrum für Informatikunterricht  
Bernadette Spieler: Pädagogische Hochschule Zürich

Die Aufgabenauswahl wurde erstellt in Zusammenarbeit mit den Organisatoren von Bebras in Deutschland, Österreich, Ungarn, Slowakei und Litauen. Besonders danken wir:

Valentina Dagienė, Tomas Šiaulys, Vaidotas Kinčius: Bebras.org

Wolfgang Pohl, Hannes Endreß, Ulrich Kiesmüller, Kirsten Schlüter, Michael Weigend: Bundesweite Informatikwettbewerbe (BWINF), Deutschland

Wilfried Baumann, Liam Baumann, Anoki Eischer, Thomas Galler, Benjamin Hirsch, Martin Kandlhofer, Katharina Resch-Schobel: Österreichische Computer Gesellschaft

Gerald Futschek, Florentina Voboril: Technische Universität Wien

Zsuzsa Pluhár: ELTE Informatikai Kar, Ungarn

Michal Winzcer: Comenius University, Slowakei

Die Online-Version des Wettbewerbs wurde auf [cuttle.org](http://cuttle.org) realisiert. Für die gute Zusammenarbeit danken wir:

Eljakim Schrijvers, Justina Dauksaite, Arne Heijenga, Dave Oostendorp, Andrea Schrijvers, Alieke Stijf, Kyra Willekes: [cuttle.org](http://cuttle.org), Niederlande

Chris Roffey: UK Bebras Administrator, Vereinigtes Königreich

Für den Support während den Wettbewerbswochen danken wir:

Hanspeter Erni: Schulleitung Sekundarschule Rickenbach

Christoph Frei: Chragokyberneticks (Logo Informatik-Biber Schweiz)

Dr. Andrea Leu, Maggie Winter, Brigitte Manz-Brunner: Senarclens Leu + Partner AG

*Diese Broschüren sind dem Andenken an Martin Guggisberg gewidmet.*

Die deutschsprachige Fassung der Aufgaben wurde ähnlich auch in Deutschland und Österreich verwendet.

Die französischsprachige Übersetzung wurde von Elsa Pellet und die italienischsprachige Übersetzung von Christian Giang erstellt.



**INFORMATIK-BIBER SCHWEIZ**  
**CASTOR INFORMATIQUE SUISSE**  
**CASTORO INFORMATICO SVIZZERA**

Der Informatik-Biber 2021 wurde vom Schweizerischen Verein für Informatik in der Ausbildung SVIA durchgeführt und von der Hasler Stiftung unterstützt.

## HASLERSTIFTUNG

Dieses Aufgabenheft wurde am 24. August 2022 mit dem Textsatzsystem  $\text{\LaTeX}$  erstellt. Wir bedanken uns bei Christian Datzko für die Entwicklung und langjährige Pflege des Systems zum Generieren der 36 Versionen dieser Broschüre (nach Sprachen und Schulstufen). Das System wurde analog zum Vorgänger-System neu programmiert, welches ab 2014 gemeinsam mit Ivo Blöchliger entwickelt wurde. Jean-Philippe Pellet danken wir für die Entwicklung der `bebras` Toolchain, die seit 2020 für die automatisierte Konvertierung der Markdown- und YAML-Quelldokumente verwendet wird.

Hinweis: Alle Links wurden am 1. Dezember 2021 geprüft.



Die Aufgaben sind lizenziert unter einer Creative Commons Namensnennung – Nicht-kommerziell – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz. Die Autoren sind auf S. 27 genannt.



# Vorwort

Der Wettbewerb «Informatik-Biber», der in verschiedenen Ländern der Welt schon seit mehreren Jahren bestens etabliert ist, will das Interesse von Kindern und Jugendlichen an der Informatik wecken. Der Wettbewerb wird in der Schweiz in Deutsch, Französisch und Italienisch vom Schweizerischen Verein für Informatik in der Ausbildung SVIA durchgeführt und von der Hasler Stiftung im Rahmen des Förderprogramms FIT in IT unterstützt.

Der Informatik-Biber ist der Schweizer Partner der Wettbewerbs-Initiative «Bebras International Contest on Informatics and Computer Fluency» (<https://www.bebas.org/>), die in Litauen ins Leben gerufen wurde.

Der Wettbewerb wurde 2010 zum ersten Mal in der Schweiz durchgeführt. 2012 wurde zum ersten Mal der «Kleine Biber» (Stufen 3 und 4) angeboten.

Der Informatik-Biber regt Schülerinnen und Schüler an, sich aktiv mit Themen der Informatik auseinander zu setzen. Er will Berührungsängste mit dem Schulfach Informatik abbauen und das Interesse an Fragenstellungen dieses Fachs wecken. Der Wettbewerb setzt keine Anwenderkenntnisse im Umgang mit dem Computer voraus – ausser dem «Surfen» im Internet, denn der Wettbewerb findet online am Computer statt. Für die Fragen ist strukturiertes und logisches Denken, aber auch Phantasie notwendig. Die Aufgaben sind bewusst für eine weiterführende Beschäftigung mit Informatik über den Wettbewerb hinaus angelegt.

Der Informatik-Biber 2021 wurde in fünf Altersgruppen durchgeführt:

- Stufen 3 und 4 («Kleiner Biber»)
- Stufen 5 und 6
- Stufen 7 und 8
- Stufen 9 und 10
- Stufen 11 bis 13

In den Altersklassen 3 und 4 hatten 9 Aufgaben zu lösen, nämlich aus den drei Schwierigkeitsstufen leicht, mittel und schwer jeweils drei. Für die Altersklassen 5 und 6 waren es je vier Aufgaben aus jeder Schwierigkeitsstufe, also 12 insgesamt. Für die restlichen Altersklassen waren es 15 Aufgaben, nämlich fünf Aufgaben pro Schwierigkeitsstufe.

Für jede richtige Antwort wurden Punkte gutgeschrieben, für jede falsche Antwort wurden Punkte abgezogen. Wurde die Frage nicht beantwortet, blieb das Punktekonto unverändert. Je nach Schwierigkeitsgrad wurden unterschiedlich viele Punkte gutgeschrieben beziehungsweise abgezogen:

	leicht	mittel	schwer
richtige Antwort	6 Punkte	9 Punkte	12 Punkte
falsche Antwort	−2 Punkte	−3 Punkte	−4 Punkte



Dieses international angewandte System zur Punkteverteilung soll den Anreiz zum blossen Erraten der Lösung eliminieren.

Jede Teilnehmerin und jeder Teilnehmer hatte zu Beginn 45 Punkte («Kleiner Biber»: 27 Punkte, Stufen 5 und 6: 36 Punkte) auf dem Punktekonto.

Damit waren maximal 180 Punkte («Kleiner Biber»: 108 Punkte, Stufen 5 und 6: 144 Punkte) zu erreichen, das minimale Ergebnis betrug 0 Punkte.

Bei vielen Aufgaben wurden die Antwortalternativen am Bildschirm in zufälliger Reihenfolge angezeigt. Manche Aufgaben wurden in mehreren Altersgruppen gestellt.

## **Für weitere Informationen:**

SVIA-SSIE-SSII Schweizerischer Verein für Informatik in der Ausbildung

Informatik-Biber

Nora A. Escherle

<https://www.informatik-biber.ch/de/kontaktieren/>

<https://www.informatik-biber.ch/>



# Inhaltsverzeichnis

Mitarbeit Informatik-Biber 2021 . . . . .	i
Vorwort . . . . .	iii
Inhaltsverzeichnis . . . . .	v
1. Mikas Stempel . . . . .	1
2. Richtiges Trikot . . . . .	5
3. Brückenbau . . . . .	7
4. Lieblingsgeschenk . . . . .	9
5. Schlüsselanhänger . . . . .	11
6. Baum fällt! . . . . .	15
7. Schildkrötenpfad . . . . .	17
8. Wasser auf die Mühle . . . . .	21
9. Kugelspiel . . . . .	25
A. Aufgabenautoren . . . . .	27
B. Sponsoring: Wettbewerb 2021 . . . . .	28
C. Weiterführende Angebote . . . . .	31







# 1. Mikas Stempel

Mika hat vier Stempel mit verschiedenen Figuren. Jeden nimmt sie einmal in die Hand und stempelt damit zweimal. Es entsteht folgendes Bild:



*Welchen Stempel hat Mika zuerst verwendet?*

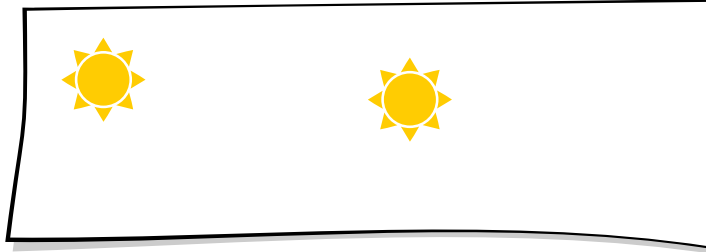




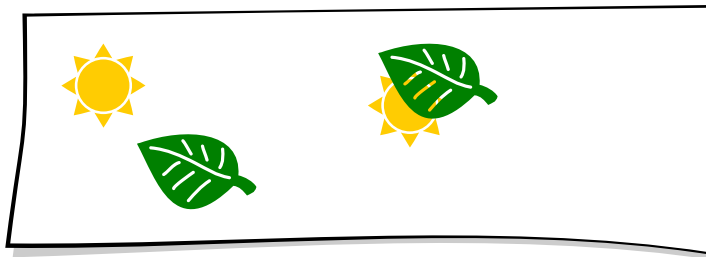
## Lösung

Die richtige Antwort ist C: 

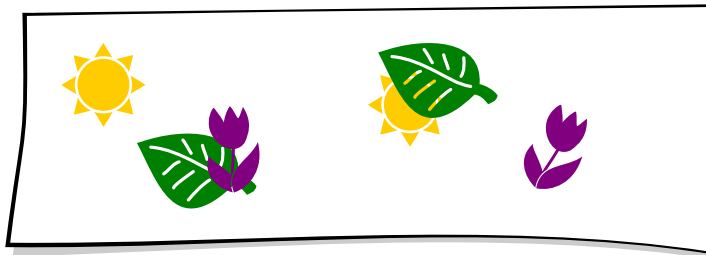
Die Reihenfolge, in der Mika die Stempel verwendet, erkennt man daran, welche Figuren andere Figuren überlagern oder von anderen Figuren überlagert werden. Die Sonne wird von allen anderen Figuren überlagert. Mika hat somit den Stempel mit der Sonne zuerst verwendet.



Das Blatt überlagert die Sonne, aber wird sowohl von der Blume als auch von dem Haus überlagert. Mika hat somit den Stempel mit dem Blatt als zweites verwendet:



Die Blume überlagert wiederum sowohl das Blatt also auch die Sonne.



Es ist jedoch nicht möglich, dass Mika den Stempel mit der Blume zuletzt verwendet hat: An zwei Stellen wird die Blume klar sichtbar von dem Haus überlagert. Somit hat Mika den Stempel mit der Blume als drittes verwendet und den Stempel mit dem Haus zuletzt.





## Dies ist Informatik!

Mikas Bild ist gewissermaßen ein Abbild oder eine *Modellierung* der Realität anhand von gestempelten Figuren auf vier verschiedenen Ebenen: eine Ebene mit Sonnen, eine Ebene mit Blättern, eine Ebene mit Blumen und eine Ebene mit Häusern. Die Überlagerung der Figuren verschiedener Ebenen ermöglicht es Mika, auf einer (zweidimensionalen) Fläche von Papier die Illusion von Tiefe und (dreidimensionaler) Räumlichkeit zu erzeugen.

Bei einer Modellierung werden üblicherweise nur Aspekte abgebildet, die notwendig sind, um eine bestimmte Aufgabe oder Funktion zu erfüllen. Die Realität wird also vereinfacht dargestellt. Modellierung ist ein wichtiges Prinzip in der Informatik.

So erstellen Informatiker:innen beispielsweise manche Computerprogramme mit dem Zweck, Teile der realen Welt möglichst schnell und präzise zu untersuchen und Erkenntnisse über diese zu gewinnen. Um solch ein Programm zu erstellen, müssen sich Informatiker:innen immer erst ein Gedankenmodell überlegen, welches die für sie wesentlichen Elemente der realen Welt enthält. Dieses Gedankenmodell ermöglicht es, ein reales Modell in Form einer Skizze oder eines Computerprogramms zu erstellen. Mit Hilfe dieses Computerprogramms, also einer Modellierung eines Teils der realen Welt, können nun Erkenntnisse über die reale Welt erlangt werden.

## Stichwörter und Webseiten

- Modellierung: [https://files.ifi.uzh.ch/rerg/amadeus/teaching/courses/infII\\_-ss04/kapitel\\_01.pdf](https://files.ifi.uzh.ch/rerg/amadeus/teaching/courses/infII_-ss04/kapitel_01.pdf)
- Ebenen, Bildbearbeitung: <https://de.wikipedia.org/wiki/Ebenentechnik>





## 2. Richtiges Trikot

Anne packt ihre Tasche, um an den Match zu gehen. Heute muss sie das Trikot mit hellen Ärmeln und schwarzem Kragen, aber ohne Streifen mitbringen.



Welches Trikot packt sie in ihre Tasche?





## Lösung



Antwort B ist richtig.



und



Das Trikot A und D passen heute nicht, weil sie schwarze Ärmel haben, und schwarz ist nicht hell.



Das Trikot C hat Streifen, deswegen passt es nicht für den heutigen Match.

Das Trikot B ist perfekt für heute: Es hat helle Ärmel und einen schwarzen Kragen und keine Streifen.

## Dies ist Informatik!

In dieser Biberaufgabe solltest du aus einer Menge von Dingen das Ding heraussuchen, das bestimmte *Bedingungen* erfüllt oder eben nicht erfüllt.

Mehrere *Teilbedingungen*, wie zum Beispiel die Farbe der Ärmel und das Stoffmuster wurden hier definiert und zu einer *Gesamtbedingung* verknüpft. Informatiker verwenden für diese Verknüpfungen *logische Operatoren*.

Wenn alle Teilbedingungen gleichzeitig erfüllt sein müssen, verwendet man den Operator *UND*: «Die Farbe der Ärmel ist hell.» *UND* «die Farbe des Kragens ist schwarz.» Wenn es reicht, wenn mindestens eine von mehreren Teilbedingungen erfüllt ist, verwendet man den Operator *ODER*. Wenn eine Teilbedingung nicht erfüllt sein darf, kann man den Operator *NICHT* verwenden, zum Beispiel: *NICHT*(Das Trikot hat Streifen.)

Für das Suchen in Datenbanken kann man mit Hilfe von Abfragesprachen sehr komplexe Bedingungen formulieren.

Die Bedingungen selbst müssen dafür klar definiert sein. Die Bedingung, dass die Ärmel hell sein müssen, kann zum Beispiel unklar sein. In einem solchen Fall schreiben Informatiker ein Programm oder eine Funktion, welche überprüft, ob eine Farbe hell ist oder nicht. Hierfür benötigen die Informatiker eine klare Definition, wann genau eine Farbe «hell» ist, sonst ist es nicht möglich, ein funktionierendes Programm zu schreiben.

## Stichwörter und Webseiten

- Boolesche Algebra: [https://de.wikipedia.org/wiki/Boolesche\\_Algebra](https://de.wikipedia.org/wiki/Boolesche_Algebra)
- Logische Verknüpfungen: [https://de.wikipedia.org/wiki/Logische\\_Verknüpfung](https://de.wikipedia.org/wiki/Logische_Verknüpfung)



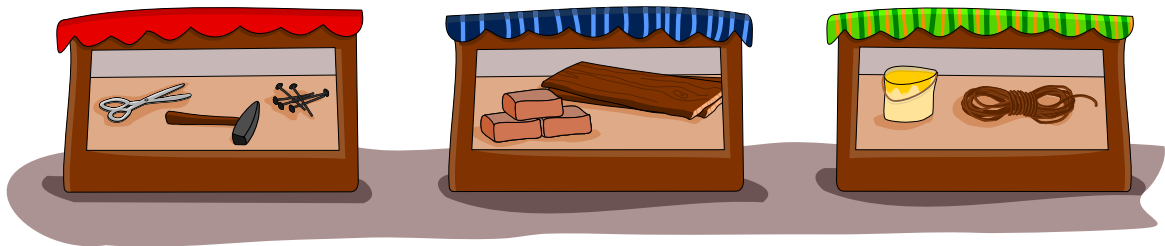
### 3. Brückenbau

Bella möchte eine Brücke bauen, über einen Bach. Sie braucht: Hammer, Nägel, Bretter und ein Seil. Im Keller findet sie einen Hammer und ein Seil.



Die anderen Sachen muss sie einkaufen. Unten siehst du drei Geschäfte, und was sie verkaufen.

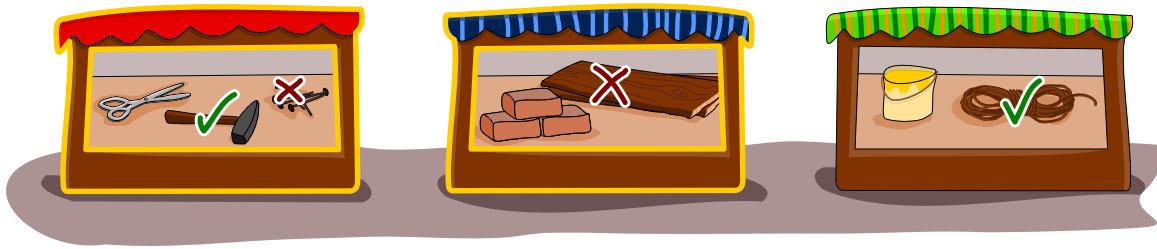
Wo kann Bella die anderen Sachen einkaufen?





## Lösung

So ist es richtig:



## Dies ist Informatik!

Die Geschäfte in dieser Biberaufgabe verkaufen insgesamt sieben Sachen: Schere, Hammer, Nägel, Ziegel, Bretter, Seil und Eimer. Das ist eine ganze Menge! Die beiden Sachen, die Bella einkaufen muss, sind ein *Teil dieser Menge*. Das kann man so malen: Man zeigt alle Sachen der gesamten Menge und markiert für jede Sache, ob sie zu Bellas Einkaufs-Teilmenge gehört oder nicht:



Genauso kann man malen, was die Geschäfte verkaufen, zum Beispiel das Geschäft links:



So sieht man auf den ersten Blick, was Bella im Geschäft links einkaufen kann: Die Nägel haben einen grünen Haken in der Einkaufs-Teilmenge und in der Verkaufs-Teilmenge.

Auch Computerprogramme müssen häufig Mengen vergleichen. Das können sie so machen wie oben gezeigt: Für jede Sache, die vorkommen kann, wird ein *Bit* benötigt. In einem Bit kann ein Computer einen von zwei Werten speichern, wie «ja» und «nein». In diesem Fall wird gespeichert, ob die Sache zu einer Menge gehört («ja») oder nicht («nein»). Dann kann ein Programm zwei Mengen so vergleichen: Es prüft, ob das Bit für eine Sache in der einen Menge «ja» ist und das Bit für die gleiche Sache in der anderen Menge auch «ja» ist. Eine solche Prüfung zweier Bits kann ein Computer besonders schnell erledigen. In der Informatik ist die Beschreibung von Mengen mit Bits deshalb sehr verbreitet.

## Stichwörter und Webseiten

- Menge: [https://de.wikipedia.org/wiki/Menge\\_\(Datenstruktur\)](https://de.wikipedia.org/wiki/Menge_(Datenstruktur))
- Bits: <https://de.wikipedia.org/wiki/Bit>
- Bitstrings: <https://de.wikipedia.org/wiki/Bitkette>





## 4. Lieblingsgeschenk

Die Biberfamilie hat drei Geschenke für ihre drei Kinder. Jedes Kind nennt zuerst sein Lieblingsgeschenk und dann das zweitliebste. Die Geschenke sollen richtig zugeteilt werden:

1. Möglichst viele Kinder sollen ihr Lieblingsgeschenk bekommen.
2. Die übrigen sollen das zweitliebste bekommen.

*Gib den Kindern die richtigen Geschenke.*



1:  , 2: 



1:  , 2: 

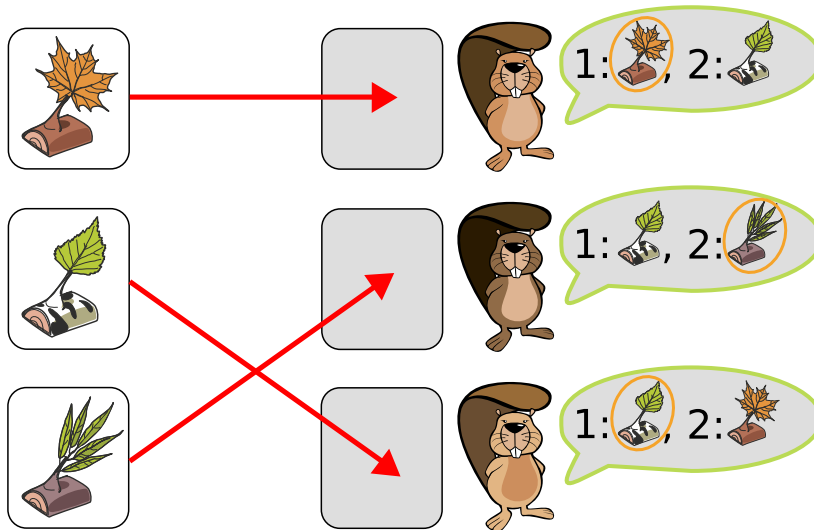


1:  , 2: 



## Lösung

Hier ist die einzige Zuordnung der Geschenke, die beide Bedingungen erfüllt.



Das dritte Geschenk wünscht sich nur der zweite Biber, deshalb muss er es bekommen. Ansonsten würde ein anderer etwas bekommen, das weder das Lieblingsgeschenk noch das zweitliebste ist. Für die anderen beiden Geschenke ist die Aufteilung dann klar: Jeder kann sein Lieblingsgeschenk bekommen.

## Dies ist Informatik!

Bei dieser Aufgabe handelt sich um ein eindeutiges *Zuordnungsproblem*: Wir möchten die Geschenke so zuordnen, dass alle Kinder ein Geschenk bekommen und es kein Kind ohne Geschenk gibt. Dabei haben die Kinder nicht nur einen einzelnen Wunsch, sondern geben eine Reihenfolge von Vorlieben an. Solche Zuordnungsprobleme mit Reihenfolgen von Vorlieben können sehr kompliziert werden. Die Informatik hilft uns dabei, solche Problem möglichst rasch zu lösen.

Eine Möglichkeit ist den Zuordnungen einen Wert zu geben: Das Lieblingsgeschenk hat Wert 1 und das zweitliebste Geschenk den Wert 2. Wir möchten den Gesamtwert minimieren. Eine *Zuordnung* (engl. *Matching*) ist *optimal*, wenn es keine andere Zuordnung mit mehr erfüllten Erstplatzierungen gibt. In der Informatik wird so eine Zuordnung als *Rang-Maximal-Matching* bezeichnet. Es gibt viele Matching-Probleme. Eines davon wird als *Problem der stabilen Paarung* (engl. *Stable Marriage Problem*) bezeichnet. Klingt interessant? Informatik ist eben ein ganz vielseitiges Fach.

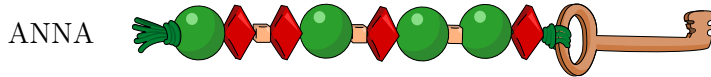
## Stichwörter und Webseiten

- Zuordnungsproblem: <https://de.wikipedia.org/wiki/Zuordnungsproblem>
- Rang-Wahl: <https://de.wikipedia.org/wiki/Rang-Wahl>

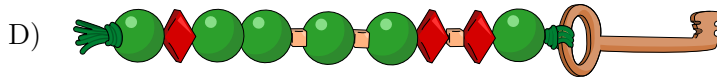
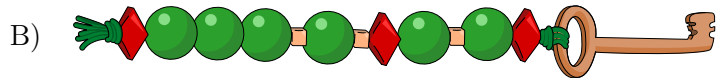
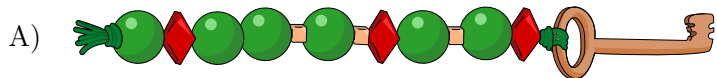


## 5. Schlüsselanhänger

ANNA, BELLA und LENA machen Schlüsselanhänger mit ihren Namen. Für die Buchstaben verwenden sie zwei Sorten von Perlen: ● und ◆. Einzelne Buchstaben trennen sie mit dieser Perle: ■.







Welchen Schlüsselanhänger hat LENA gemacht?




















## Lösung

Die richtige Antwort ist A) 



Das Wort LENA beginnt mit L. In BELLA ist L der dritte und vierte Buchstabe und wir erkennen in BELLA für L die Perlenfolge . Nur Antworten A) und D) beginnen mit diesem Buchstaben und können die Lösung sein. Der zweite Buchstabe von LENA, also E, ist auch der zweite von BELLA, dort finden wir im Anhänger jeweils die Perle . Sowohl bei A) als auch bei D) haben wir  als zweiten Buchstaben, also könnten sie beide weiterhin die Lösung sein. Als nächstes wollen wir die Perlen für den Buchstaben N herausfinden. In ANNAs Anhänger finden wir für N die Perlenfolge . Und das sind nur in der Lösung A) die nächsten Perlen.

Eine andere Möglichkeit, die Perlen für LENAs Anhänger herauszufinden, ist es, eine Tabelle mit den Perlen für die bereits bekannten Buchstaben zu erstellen. Von ANNAs Anhänger finden wir für A die Perlenfolge  und für N . Von BELLAs Anhänger finden wir die Perlenfolge für B: , für E:  und L: .

Buchstaben	Perlen
A	
N	
B	
E	
L	

Daher können wir den Anhänger für LENA aus den Perlen ,  und  formen, wenn wir zusätzlich die einzelnen Buchstaben mit der Perle  trennen. So erhalten wir den Anhänger , was die Antwort A) ist. Wenn wir die anderen gegebenen Anhänger mit Hilfe der Codierungstabelle entschlüsseln, finden wir für B) BENA, C) NENA und für D) LEAN.

## Dies ist Informatik!

Information wird *codiert*, um Nachrichten unter bestimmten Bedingungen übertragen zu können oder um die Information geheim (*verschlüsselt*) zu übertragen. In dieser Aufgabe basiert die Codierung auf dem Morsecode. Der Punkt • des Morsecode wird durch die runde Perle  dargestellt und der Strich – durch . Der Buchstabe A ist im Morsecode •–, also im Perlencode . Um beliebige Texte zu codieren, brauchen wir Codes für alle Buchstaben des Alphabets. Der Morsecode entstand im 19. Jahrhundert. Samuel Morse erfand 1837 einen einfachen elektromagnetischen Schreibtelegraphen. Der verwendete Code umfasste damals nur die zehn Ziffern 0 bis 9; die übertragenen Zahlen mussten mit Hilfe einer Tabelle in Buchstaben und Wörter übersetzt werden. Alfred Lewis Vail, ein Mitarbeiter von Morse, entwickelte ab 1838 den ersten Code, der auch Buchstaben umfasste. Der Code wurde entwickelt, um Texte akustisch, optisch oder elektrisch über Telegraphenleitungen zu übertragen. Ein Punkt ist eine kurze Übertragungsdauer und ein Strich eine dreimal so lange. Zwischen den Buchstaben muss eine Pause eingehalten werden. Eine längere Pause trennt Wörter. Noch heute wird



der Morsecode für das Notsignal SOS verwendet. SOS im Morsecode **••• — — — •••** (3x kurz, 3x lang, 3x kurz) kann sehr leicht durch Rufen, Klopfen, oder mit der Taschenlampe übertragen werden. In der elektronischen Datenverarbeitung werden Zeichen über einen Zahlenwert codiert, um sie zu übertragen oder zu speichern.

## Stichwörter und Webseiten

- Zeichenkodierung: <https://de.wikipedia.org/wiki/Zeichenkodierung>
- Morsecode: <https://de.wikipedia.org/wiki/Morsecode>





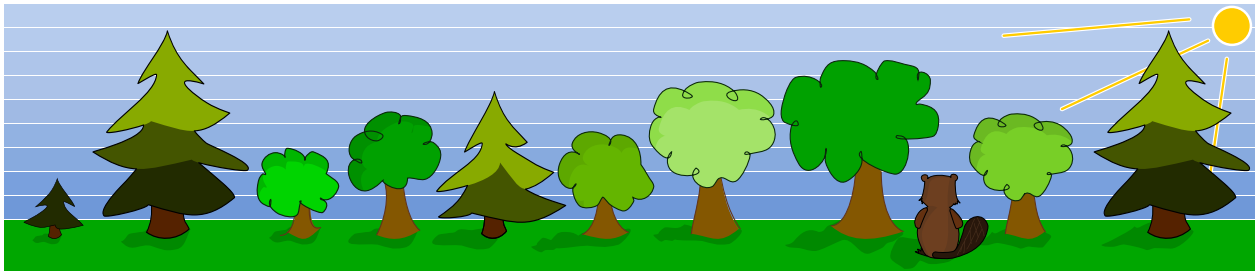
## 6. Baum fällt!

Ein Biber möchte einen Damm bauen. Damit er immer die richtigen Bäume fällt, hat er sich zwei Bedingungen überlegt:

Er fällt einen Baum nur dann, wenn ...

- ... direkt links daneben ein kleinerer Baum wächst und ...
- ... direkt rechts daneben ein grösserer Baum wächst.

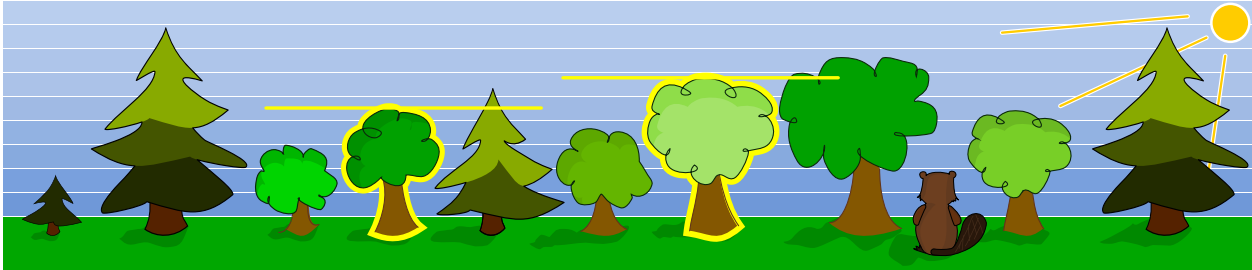
*Welche Bäume wird der Biber fällen?*





## Lösung

Nur die Bäume an vierter Stelle und siebter Stelle erfüllen die vorgegebenen Bedingungen: Direkt links befindet sich ein kleiner Baum UND direkt rechts ein grösserer Baum.



## Dies ist Informatik!

In der Informatik geht es oft darum, Probleme zu lösen, die durch eine Reihe von logischen *Einschränkungen* (engl. *constraints*) spezifiziert sind. Die Aufgabe besteht darin, eine Lösung zu finden, die alle gegebenen Einschränkungen erfüllt. Komplexere Aufgaben als diese können behandelt werden, indem die Einschränkungen mit Hilfe von *logischen Operatoren* kombiniert werden. Die Konjunktion ( $\wedge$  oder auch UND-Operator) beispielsweise liefert im Ausdruck  $A \wedge B$  genau dann «wahr» als Ergebnis, wenn die beiden Einschränkungen ebenfalls wahr sind. In dieser Aufgabe wäre das dann also: «Der Baum links ist kleiner.»  $\wedge$  «Der Baum rechts ist grösser.» Dieses grundlegende Prinzip kommt in nahezu allen Bereichen der Informatik vor, wie beispielsweise bei vielen Sortier-Algorithmen, zum Beispiel dem *Bubble Sort*, hier werden immer mehrere Objekte einer Liste auf gewisse Einschränkungen überprüft, um sie danach, sofern notwendig, an eine andere Position in der Liste zu verschieben. Dieses Prinzip wird solange wiederholt, bis die Liste vollständig sortiert ist.

## Stichwörter und Webseiten

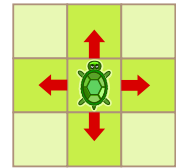
- Algorithmisches Denken (algorithmic thinking)
- logische Operatoren: [https://de.wikipedia.org/wiki/Logischer\\_Operator](https://de.wikipedia.org/wiki/Logischer_Operator)
- Sortieren: <https://sorting.at/>
- Constraint Satisfaction Problem:  
<https://de.wikipedia.org/wiki/Constraint-Satisfaction-Problem>





## 7. Schildkrötenpfad

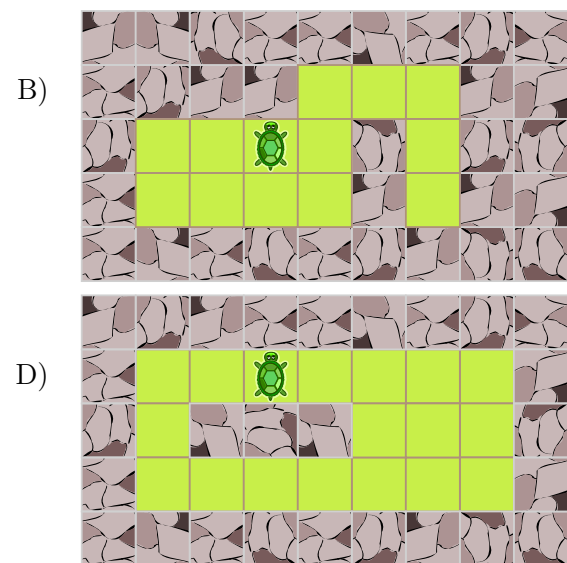
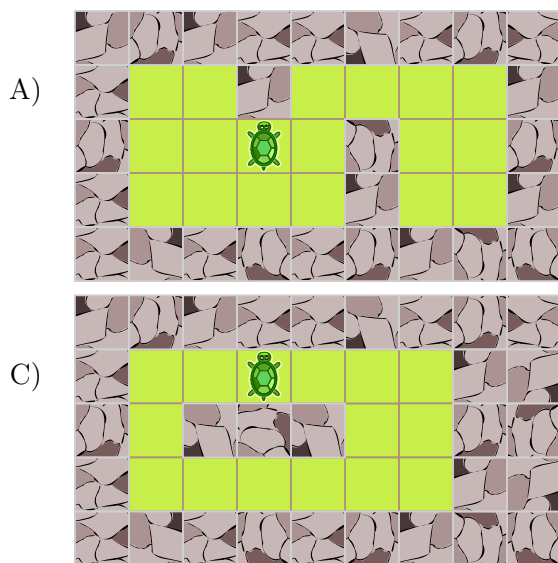
Eine Schildkröte soll verschiedene Gärten abgrasen. Jeder Garten ist in Quadrate unterteilt, die entweder mit Gras oder Steinen bedeckt sind. Die Schildkröte kann keine Steine überqueren. Sie kann sich aber von einem Grasfeld zu einem anderen Grasfeld direkt daneben bewegen.

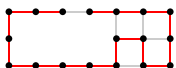


Die Schildkröte soll die Gärten vollständig abgrasen. Sie startet auf dem Feld, auf dem sie im Bild steht. Am Ende soll sie in jedem Grasfeld genau einmal gewesen sein.

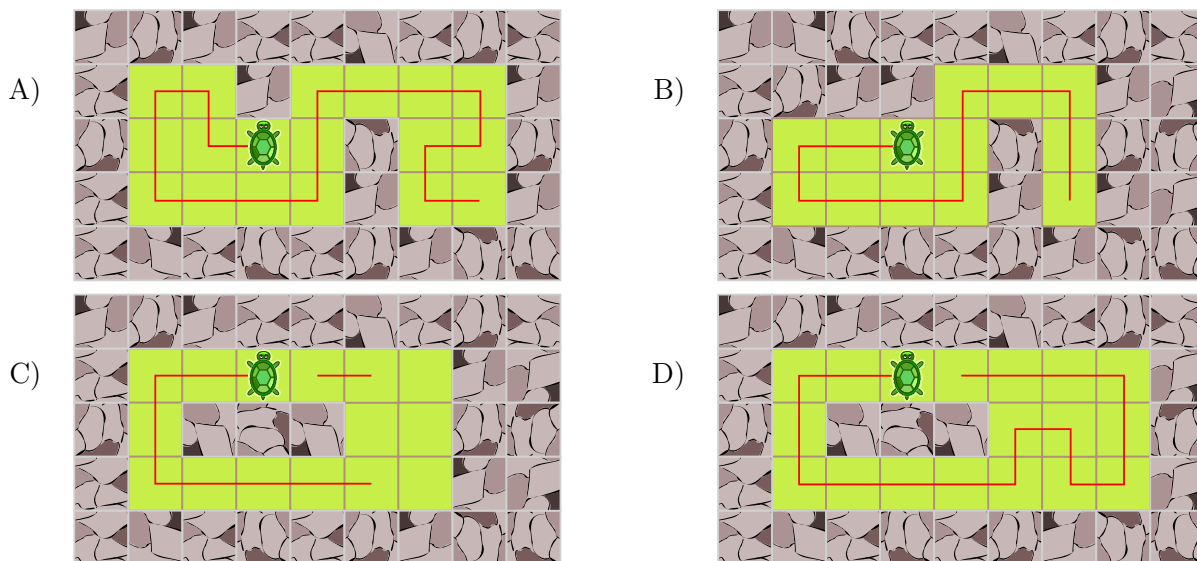
Leider kann die Schildkröte so einen der Gärten nicht vollständig abgrasen.

*Welcher ist es?*





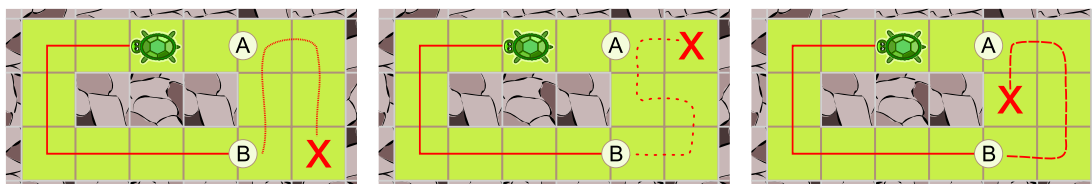
## Lösung



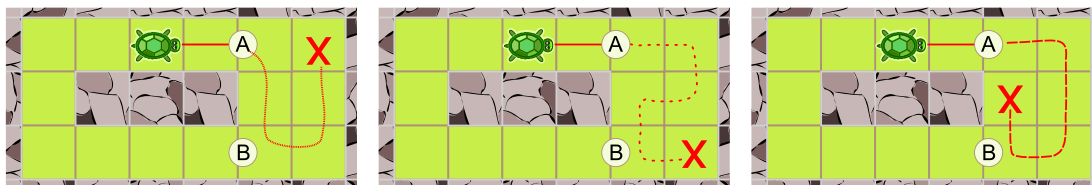
Die Gärten A, B und D kann die Schildkröte vollständig abgrasen.

Den Garten C kann die Schildkröte so nicht abgrasen. Die Schildkröte hat von ihrem Startpunkt aus nur 2 Möglichkeiten:

- Geht sie zuerst nach links, kommt sie zu Punkt B. Von dort müsste sie die 6 Felder rechts so abgrasen, dass sie am Ende Punkt A erreicht. Aber keiner der von B aus möglichen Pfade endet bei A.



- Geht die Schildkröte zuerst nach rechts, kommt sie zu A und müsste die 6 Felder so abgrasen, dass sie am Ende Punkt B erreicht. Jetzt kann man gleich argumentieren wie vorher, man muss nur oben und unten vertauschen. Es gibt also auch so keinen geeigneten Pfad.

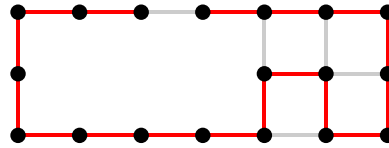


## Dies ist Informatik!

Die Schildkröte soll einen Weg durch ihren Garten finden und dabei jedes Grasfeld genau einmal besuchen. Das Problem hinter dieser Aufgabe ist ein sogenanntes *Hamiltonpfadproblem*.



Ein Schildkrötengarten (also die Grasquadrate) kann so betrachtet werden: Jedes Grasquadrat ist ein *Knoten* (dargestellt als Punkt). Der Garten D sieht dann so aus:



Für derartige Strukturen (von Informatikern, aber auch von Mathematikern *Graphen* genannt) stellte sich im 19. Jahrhundert Sir William Rowan Hamilton die Frage, ob es einen Pfad entlang der Kanten gibt, der jeden Knoten genau einmal besucht. Einen solchen Pfad nennt man deshalb auch *Hamilton-Pfad*. Die Frage, ob es überhaupt einen Hamilton-Pfad gibt, ist im Allgemeinen sehr schwer zu lösen. Niemand kennt einen *Algorithmus*, der für beliebige Graphen effizient (in mehr oder weniger nützlicher Frist) entscheiden könnte, ob es im vorgegebenen Graphen einen Hamilton-Pfad gibt oder nicht. Wir wissen auch nicht, ob es einen solchen effizienten Algorithmus überhaupt geben kann. Dies gilt für alle sogenannten *NP-vollständigen Probleme*, wovon das Hamilton-Pfad-Problem eines der berühmtesten ist.

## Stichwörter und Webseiten

- Graph, Hamiltonpfad: <https://de.wikipedia.org/wiki/Hamiltonkreisproblem>



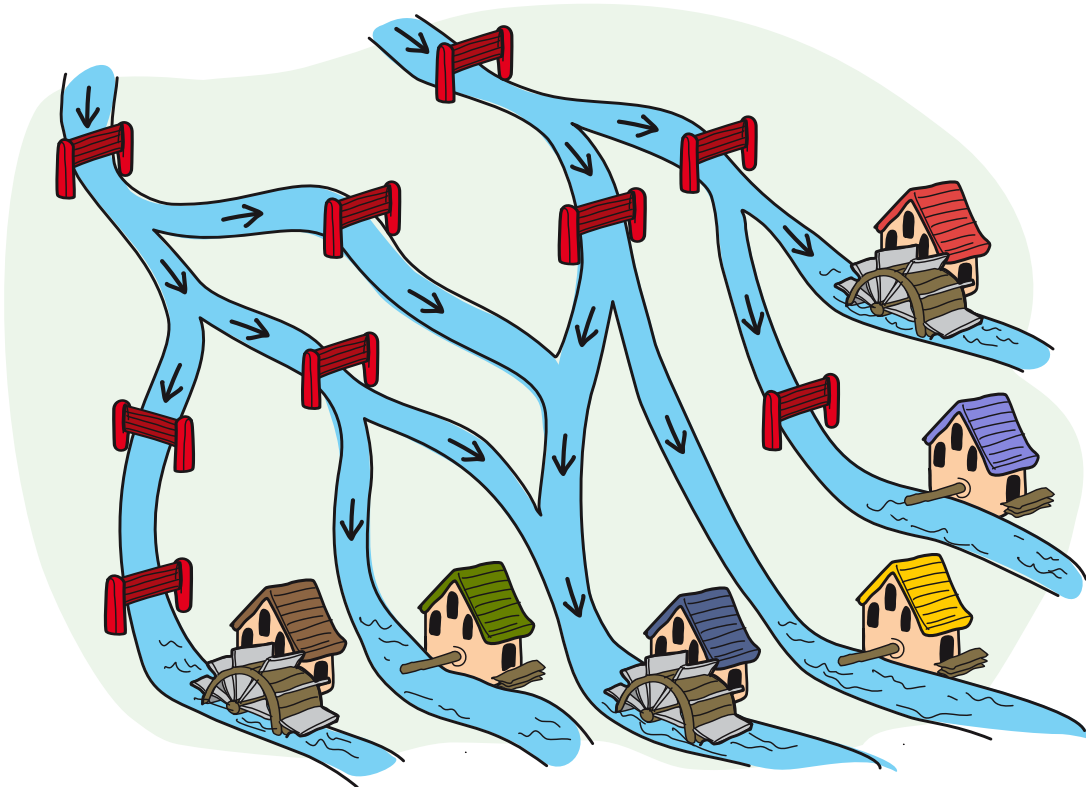


## 8. Wasser auf die Mühle

Müller Mert hat sechs Mühlen. Bei drei davon muss er noch das Mühlrad einbauen. Dafür darf kein Wasser mehr zu diesen Mühlen fließen. Zu den anderen Mühlen soll aber weiterhin Wasser fließen.

Das Wasser kann nur nach unten fließen. Ein geschlossener Schieber stoppt das Wasser.

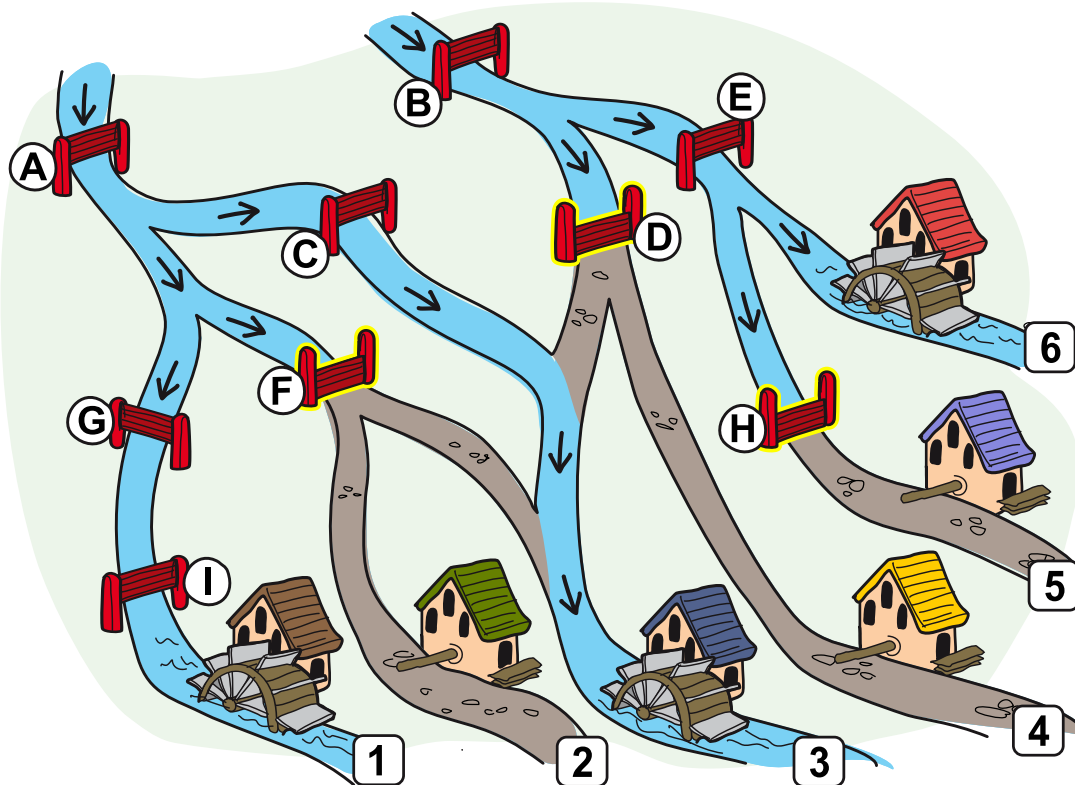
*Welche Schieber soll Mert schliessen?*





## Lösung

Die richtige Antwort ist: Es sind drei Schieber zu schliessen, die in der folgenden Zeichnung mit D, F und H bezeichnet wurden.



Dies ist die einzige Möglichkeit, mit der kein Wasser mehr zu den Mühlen 2, 4 und 5 ohne Mühlrad fließt, während die Mühlen 1, 3 und 6 weiter Wasser bekommen:

- Die Schieber A, G und I müssen alle offen bleiben, da sonst kein Wasser mehr zur Mühle 1 fließen würde.
- Die Schieber B und E müssen ebenfalls geöffnet bleiben, da sonst kein Wasser mehr zur Mühle 6 fließen würde.
- Weil die Schieber B und E offen bleiben, muss der Schieber H geschlossen werden, da sonst Wasser zur Mühle 5 fließen würde.
- Weil der Schieber A offen bleibt, muss der Schieber F geschlossen werden, da sonst Wasser zur Mühle 2 fließen würde.
- Weil der Schieber B offen bleibt, muss der Schieber D geschlossen werden, da sonst Wasser zur Mühle 4 fließen würde.
- Weil die Schieber D und F geschlossen werden, muss der Schieber C offen bleiben, da sonst kein Wasser mehr zur Mühle 3 fließen würde.



## Dies ist Informatik!

In dieser Aufgabe wird das Fließen des Wassers zu den Mühlen durch *Bedingungen* geregelt. Zum Beispiel fließt genau dann Wasser zur Mühle 6, wenn die beiden Schieber B und E offen stehen. Und hier noch ein zweites, etwas komplizierteres Beispiel: Zur Mühle 3 fließt genau dann Wasser, wenn mindestens eine oder gleich beide der folgenden Bedingungen erfüllt sind:

- Der Schieber A ist offen und einer der beiden Schieber C oder F ist offen.
- Die beiden Schieber B und D sind offen.

Solche zusammengesetzten Bedingungen werden mit den *logischen Operatoren* UND (als Symbol:  $\wedge$ ) bzw. ODER (als Symbol:  $\vee$ ) erzielt. Solche Operatoren verknüpfen Wahrheitswerte wie wahr oder falsch. Sind also A und B zwei Wahrheitswerte, so kann man angeben, welche Wahrheitswerte die zusammengesetzten Ausdrücke «A UND B» bzw. «A ODER B» haben:

A	B	A UND B	A ODER B
falsch	falsch	falsch	falsch
wahr	falsch	falsch	wahr
falsch	wahr	falsch	wahr
wahr	wahr	wahr	wahr

In der Informatik (und auch in der Mathematik) wird also die Aussage «A ODER B» auch dann als richtig angesehen, wenn beide, A und B, richtig sind. Die Aussage «Es fließt Wasser zur Mühle 6.» ist gleichbedeutend mit:

«Der Schieber B ist offen.» UND «Der Schieber E ist offen.».

Die Aussage «Es fließt Wasser zur Mühle 3.» aus unserem zweiten Beispiel ist gleichbedeutend mit:

(«Der Schieber A ist offen.» UND («Der Schieber C ist offen.» ODER «Der Schieber F ist offen.»))  
ODER («Der Schieber B ist offen.» UND «Der Schieber D ist offen.»).

Jedes UND und jedes ODER verbindet zwei Aussagen. Die Klammer machen klar, in welcher Reihenfolge die Aussagen verbunden werden. Beim Programmieren ist das korrekte Formulieren von Bedingungen wichtig. Die Verknüpfung mit den logischen Operatoren und Klammern ist nützlich, um komplexere Bedingungen zu formulieren. Sowohl bei Verzweigung mit Hilfe von `if` als auch bei `while`-Schleifen verwenden wir Bedingungen, um den Programmablauf zu steuern.

## Stichwörter und Webseiten

- Bedingte Anweisung: [https://de.wikipedia.org/wiki/Bedingte\\_Anweisung\\_und\\_Verzweigung](https://de.wikipedia.org/wiki/Bedingte_Anweisung_und_Verzweigung)
- Boolsche Variable: <https://de.wikipedia.org/wiki/Boolean>
- Boolsche Operatoren: [https://de.wikipedia.org/wiki/Boolescher\\_Operator](https://de.wikipedia.org/wiki/Boolescher_Operator)



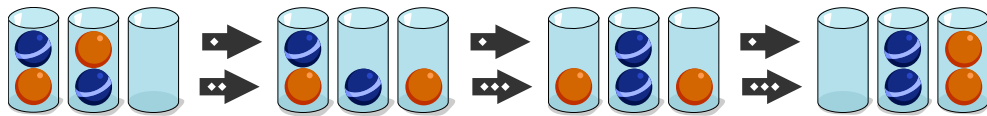




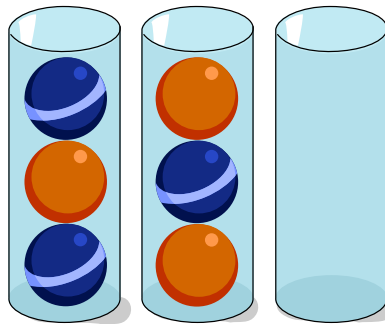
## 9. Kugelspiel

Die Biber möchten Kugeln nach ihrer Farbe ordnen. Am Ende sollen sich alle Kugeln in zwei Gläsern befinden. In einem Glas sollen die Kugeln dieselbe Farbe haben. Dabei sind diese drei Regeln zu befolgen:

- ➡ Regel 1: In einem Schritt kann nur die oberste Kugel eines Glases bewegt werden.
- ➡➡ Regel 2: Eine Kugel kann in ein leeres Glas bewegt werden.
- ➡➡➡ Regel 3: Eine Kugel kann in ein Glas bewegt werden, wenn dort noch Platz frei ist und die darunter liegende Kugel dieselbe Farbe hat.



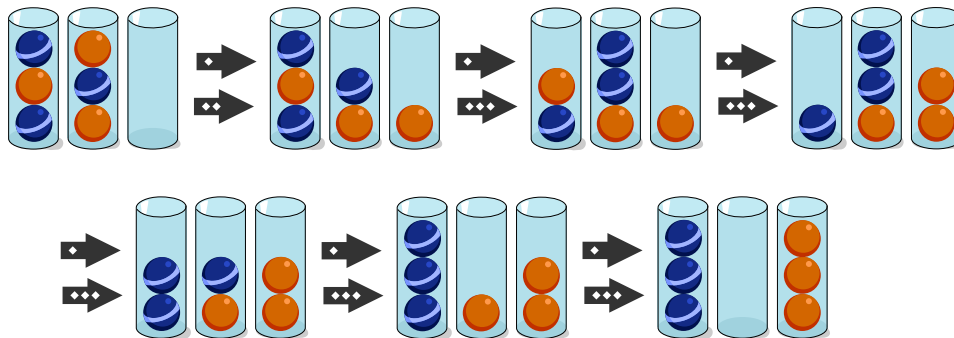
Ordne die Kugeln, indem du sie nach den drei Regeln bewegst.





## Lösung

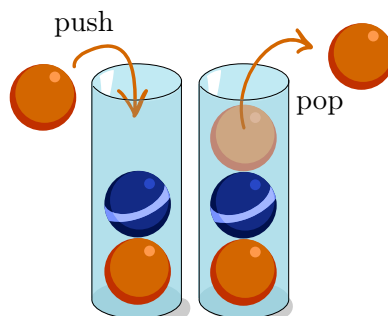
Die Kugeln können in der folgenden Reihenfolge bewegt werden:



Um die Kugeln neu zu ordnen, benötigt man mindestens 6 Schritte. Es gibt auch noch andere Möglichkeiten, die Kugeln in nur 6 Schritten neu zu ordnen.

## Dies ist Informatik!

In dieser Aufgabe bewegst du die Kugeln ähnlich wie der Computer in einem *Stapelspeicher* Daten verwaltet: Er kann nur *oben ein Element* (in der Aufgabe eine Kugel) *hinzufügen* (engl. *push*) und nur das *oberste Element wieder entfernen* (engl. *pop*).




Auf die unteren Elemente kann der Computer nur zugreifen, wenn zuerst die Kugeln oberhalb entfernt werden. Und das Element, das als letztes gespeichert wurde, wird der Computer wieder als erstes entfernen. Dieses Prinzip nennen die Informatiker *Last-in-First-out-Prinzip* (kurz *LIFO*).

## Stichwörter und Webseiten

- Stapelspeicher, Kellerspeicher: <https://de.wikipedia.org/wiki/Stapelspeicher>



## A. Aufgabenautoren

 Daumilas Ardickas

 Michael Barot

 Liam Baumann


 Wilfried Baumann

 Carmel Carroll

 Christian Datzko


 Susanne Datzko

 Nora A. Escherle


 Lidia Feklistova

 Fabian Frei


 Gerald Futschek

 Yasemin Gülbahar


 Ezgi Arzu Güneş

 Benjamin Hirsch

 Andrea Hrušecká

 Tiberiu Iorgulescu

 YongJu Jeon


 Soojin Jun


 Ungyeol Jung

 Filiz Kalelioğlu

 Martin Kandlhofer

 Dong Yoon Kim

 Jihye Kim

 Vaidotas Kinčius

 V́ictor Koleszar

 Taina Lehtimäki


 Tom Naughton

 Graciela Oyhenard

 Jean-Philippe Pellet


 Zsuzsa Pluhár


 Wolfgang Pohl

 Rosario Schunk

 Bernadette Spieler

 Troy Vasiga

 Florentina Voboril

 Kyra Willekes

 Hongjin Yeh



## B. Sponsoring: Wettbewerb 2021

### HASLERSTIFTUNG

<http://www.haslerstiftung.ch/>

Stiftungszweck der Hasler Stiftung ist die Förderung der Informations- und Kommunikationstechnologie (IKT) zum Wohl und Nutzen des Denk- und Werkplatzes Schweiz. Die Stiftung will aktiv dazu beitragen, dass die Schweiz in Wissenschaft und Technologie auch in Zukunft eine führende Stellung innehat.



<http://www.baerli-biber.ch/>

Schon in der vierten Generation stellt die Familie Bischofberger ihre Appenzeller Köstlichkeiten her. Und die Devise der Bischofbergers ist dabei stets dieselbe geblieben: «Hausgemacht schmeckt's am besten». Es werden nur hochwertige Rohstoffe verwendet: reiner Bienenhonig und Mandeln allererster Güte. Darum ist der Informatik-Biber ein «echtes Biberli».



<http://www.verkehrshaus.ch/>



Standortförderung beim Amt für Wirtschaft und Arbeit Kanton Zürich



i-factory (Verkehrshaus Luzern)

Die i-factory bietet ein anschauliches und interaktives Erproben von vier Grundtechniken der Informatik und ermöglicht damit einen Erstkontakt mit Informatik als Kulturtechnik. Im optischen Zentrum der i-factory stehen Anwendungsbeispiele zur Informatik aus dem Alltag und insbesondere aus der Verkehrswelt in Form von authentischen Bildern, Filmbeiträgen und Computer-Animationen. Diese Beispiele schlagen die Brücke zwischen der spielerischen Auseinandersetzung in der i-factory und der realen Welt.



<http://www.ubs.com/>

Wealth Management IT and UBS Switzerland IT



**OXOCARD**

<http://www.oxocard.ch/>

OXOcard: Spielend programmieren lernen  
OXON

**educaTEC**

<https://educatec.ch/>

educaTEC

Wir sind MINT-Experten. Seit unserer Gründung 2004 verfolgen wir das Ziel, Technik und ingenieurwissenschaftliches Denken in öffentlichen und privaten Schulen der Schweiz zu fördern. In Kombination mit kompetenter Beratung und Unterstützung offerieren wir Lehrkräften innovative Lehrmaterialien von weltweit führenden Herstellern sowie Lernkonzepte für den MINT-Bereich und verwandte Fächer.

**senarclens  
leu+partner**  
strategische kommunikation

<http://senarclens.com/>

Senarclens Leu & Partner

**ABZ**

AUSBILDUNGS- UND BERATUNGSZENTRUM  
FÜR INFORMATIKUNTERRICHT

<http://www.abz.inf.ethz.ch/>

Ausbildungs- und Beratungszentrum für Informatikunterricht der ETH Zürich.

**hep/** haute  
école  
pédagogique  
vaud

<http://www.hepl.ch/>

Haute école pédagogique du canton de Vaud

**PH LUZERN  
PÄDAGOGISCHE  
HOCHSCHULE**

<http://www.phlu.ch/>

Pädagogische Hochschule Luzern

**n|w** Fachhochschule  
Nordwestschweiz

<https://www.fhnw.ch/de/die-fhnw/hochschulen/ph>

Pädagogische Hochschule FHNW

Scuola universitaria professionale  
della Svizzera italiana

<http://www.supsi.ch/home/supsi.html>

La Scuola universitaria professionale della Svizzera italiana  
(SUPSI)

**SUPSI**



PÄDAGOGISCHE  
HOCHSCHULE  
ZÜRICH



<https://www.phzh.ch/>  
Pädagogische Hochschule Zürich



## C. Weiterführende Angebote

### Das Lehrmittel zum Informatik-Biber

#### Module

Verkehr – Optimieren

Musik – Komprimieren

Geheime Botschaften – Verschlüsseln

Internet – Routing

Apps

Auszeichnungssprachen

<http://informatik-biber.ch/einleitung/>

Das Lehrmittel zum Biber-Wettbewerb ist ein vom SVIA, dem schweizerischen Verein für Informatik in der Ausbildung, initiiertes Projekt und hat die Förderung der Informatik in der Sekundarstufe I zum Ziel.

Das Lehrmittel bringt Jugendlichen auf niederschwellige Weise Konzepte der Informatik näher und zeigt dadurch auf, dass die Informatikbranche vielseitige und spannende Berufsperspektiven bietet.

Lehrpersonen der Sekundarstufe I und weiteren interessierten Lehrkräften steht das Lehrmittel als Ressource zur Vor- und Nachbereitung des Wettbewerbs kostenlos zur Verfügung.

Die sechs Unterrichtseinheiten des Lehrmittels wurden seit Juni 2012 von der LerNetz AG in Zusammenarbeit mit dem Fachdidaktiker und Dozenten Dr. Martin Guggisberg der PH FHNW entwickelt. Das Angebot wurde zweisprachig (Deutsch und Französisch) entwickelt.



I learn it: <http://ilearnit.ch/>

In thematischen Modulen können Kinder und Jugendliche auf dieser Website einen Aspekt der Informatik auf deutsch und französisch selbständig entdecken und damit experimentieren. Derzeit sind sechs Module verfügbar.

010100110101011001001001  
010000010010110101010011  
010100110100100101000101  
001011010101001101010011  
010010010100100100100001

**SV!A**

[www.svia-ssie-ssii.ch](http://www.svia-ssie-ssii.ch)  
schweizerischer vereinfürinformatikind  
erausbildung//sociétésuissepourl'infor  
matique dans l'enseignement//societàsviz  
zera per l'informatica nell'insegnamento

Werden Sie SVIA Mitglied – <http://svia-ssie-ssii.ch/svia/mitgliedschaft> und unterstützen Sie damit den Informatik-Biber.

Ordentliches Mitglied des SVIA kann werden, wer an einer schweizerischen Primarschule, Sekundarschule, Mittelschule, Berufsschule, Hochschule oder in der übrigen beruflichen Aus- und Weiterbildung unterrichtet.

Als Kollektivmitglieder können Schulen, Vereine oder andere Organisationen aufgenommen werden.