



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

Aufgaben und Lösungen 2016 Schuljahre 9/10

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

Herausgeber:

Christian Datzko, Hanspeter Erni

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001

SV!A

www.svia-ssie-ssii.ch
schweizerischerverein für informatik und
erziehung // société suisse de l'inform
atique dans l'enseignement // società sviz
zera per l'informatica nell'insegnamento



Mitarbeit Informatik-Biber 2016

Andrea Adamoli, Christian Datzko, Susanne Datzko, Olivier Ens, Hanspeter Erni, Martin Guggisberg, Corinne Huck, Carla Monaco, Nicole Müller, Gabriel Parriaux, Jean-Philippe Pellet, Julien Ragot, Beat Trachsler.

Herzlichen Dank an:

Juraj Hromkovič, Giovanni Serafini, Urs Hauser, Tobias Kohn, Ivana Kosírová, Serena Pedrocchi, Björn Steffen: ETHZ

Valentina Dagienė: Bebras.org

Hans-Werner Hein, Wolfgang Pohl, Peter Rossmann: Bundesweite Informatikwettbewerbe (BWINF), Deutschland

Anna Morpurgo, Violetta Lonati, Mattia Monga: Italien

Gerald Futschek: Oesterreichische Computer Gesellschaft, Österreich

Zsuzsa Pluhár: ELTE Informatikai Kar, Ungarn

Eljakim Schrijvers, Daphne Blokhuis, Marissa Engels: Eljakim Information Technology bv, Niederlande

Roman Hartmann: hartmannGestaltung (Flyer Informatik-Biber Schweiz)

Christoph Frei: Chragokyberneticks (Logo Informatik-Biber Schweiz)

Pamela Aeschlimann, Andreas Hieber, Aram Loosmann, Daniel Vuille, Peter Zurflüh: Lernetz.ch (Webseite)

Andrea Leu, Maggie Winter, Brigitte Maurer: Senarclens Leu + Partner

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

Die französischsprachige Übersetzung wurde von Nicole Müller und die italienischsprachige Übersetzung von Andrea Adamoli erstellt.



INFORMATIK-BIBER SCHWEIZ
CASTOR INFORMATIQUE SUISSE
CASTORO INFORMATICO SVIZZERA

Der Informatik-Biber 2016 wurde vom Schweizerischen Verein für Informatik in der Ausbildung SVIA durchgeführt und von der Hasler Stiftung unterstützt.

HASLERSTIFTUNG

Hinweis: Alle Links wurden am 1. November 2016 geprüft. Dieses Aufgabenheft wurde am 13. November 2016 mit dem Textsatzsystem \LaTeX erstellt.



Die Aufgaben sind lizenziert unter einer Creative Commons Namensnennung – Nicht-kommerziell – Weitergabe unter gleichen Bedingungen 4.0 International Lizenz. Die Autoren sind auf S. 40 genannt.



Vorwort

Der Wettbewerb „Informatik-Biber“, der in verschiedenen europäischen Ländern 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“ (<http://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 Berührungängste 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“ auf dem 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 2016 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

Die Stufen 3 und 4 hatten 9 Aufgaben zu lösen, jeweils drei davon aus den drei Schwierigkeitsstufen leicht, mittel und schwer.

Jede der anderen Altersgruppen hatte 15 Aufgaben zu lösen, jeweils fünf davon aus den drei Schwierigkeitsstufen leicht, mittel und schwer.

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

Das international angewandte System zur Punkteverteilung soll dem erfolgreichen Erraten der richtigen Lösung durch die Teilnehmenden entgegenwirken.

Jede Teilnehmerin und jeder Teilnehmer hatte zu Beginn 45 Punkte (Kleiner Biber 27) auf dem Punktekonto.

Damit waren maximal 180 (Kleiner Biber: 108) 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

Hanspeter Erni

biber@informatik-biber.ch

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

 <https://www.facebook.com/informatikbiberch>



Inhaltsverzeichnis

Mitarbeit Informatik-Biber 2016	i
Vorwort	ii
1. Nachrichtensalat	1
2. Hierarchie	3
3. Brückenbau	5
4. Verkehrshüetli-Versteck	7
5. Gruppenarbeit	9
6. Bonbonnieren	13
7. Durch den Tunnel	15
8. Hände schütteln	17
9. Nim	21
10. Sortierte Trikot-Nummern	23
11. Palettenlift	25
12. Kugelspiel	29
13. Zwei Möglichkeiten	33
14. Tramkreuzung	35
15. Codierung von Flaggen	37
A. Aufgabenautoren	40
B. Sponsoring: Wettbewerb 2016	41
C. Weiterführende Angebote	44



1. Nachrichtensalat

Die Agenten Boris und Bertha kommunizieren über geheime Nachrichten, die niemand ausser ihnen verstehen soll. Boris möchte an Bertha folgende geheime Nachricht senden:

TREFFENMITBILLYUM6

Er schreibt die Buchstaben des Textes nacheinander in eine Tabelle mit vier Spalten und fünf Reihen, von links nach rechts Zeile für Zeile von oben nach unten. Wenn in der Tabelle am Ende noch Felder leer bleiben, schreibt er jeweils einen Stern hinein. Das Bild zeigt das Ergebnis.

T	R	E	F
F	E	N	M
I	T	B	I
L	L	Y	U
M	6	*	*

Dann erzeugt er die geheime Nachricht. Er schreibt die Zeichen aus der Tabelle auf einen neuen Zettel von oben nach unten, Spalte für Spalte von links nach rechts.

TFILMRETL6ENBY*FMIU*

Bertha verwendet für ihre Antwort dieselbe Methode. Sie sendet folgende geheime Nachricht an Boris:

OHDRIKWETNIEDS*CROE*

Welche Antwort hat Bertha gesendet?

- A) OKICHWERDEKOMMEN
- B) OKICHWERDEDORTSEIN
- C) WIRSTDUAUCHDASEIN
- D) OHICKANNNICHTWARTEN



Lösung

So findet man den ursprünglichen Text heraus: Man schreibt die empfangene geheime Nachricht wieder in eine Tabelle mit vier Spalten und fünf Zeilen, diesmal allerdings von oben nach unten Spalte für Spalte, links oben beginnend.

O	K	I	C
H	W	E	R
D	E	D	O
R	T	S	E
I	N	*	*

Liest man den Text von links nach rechts Zeile für Zeile, erhält man diese Nachricht:

OKICHWERDEDORTSEIN

Die Sternchen am Ende gehören nicht mehr zur Nachricht.

Dies ist Informatik!

Nachrichten, die wir über ein Computernetz senden, können leicht abgefangen werden. Wenn wir aber Nachrichten senden, die Passwörter oder private Information enthalten, möchten wir, dass nur der Empfänger den Text lesen kann und niemand anderes. In solchen Fällen kann man die Nachricht (den Klartext) verschlüsseln und in eine geheime Nachricht umwandeln. Nur der Empfänger weiss, wie man die geheime Nachricht wieder entschlüsseln und den ursprünglichen Klartext zurückgewinnen kann. Es gibt viele Verschlüsselungsverfahren.

Die Methode, die in dieser Aufgabe verwendet worden ist, nennt man Transposition. Alle Buchstaben des Klartextes bleiben erhalten. Es wird nur ihre Reihenfolge geändert. Diese Art von Verschlüsselung ist ganz einfach zu knacken, sie wurde schon vor ca. 2400 Jahren verwendet. Eigentlich ist es gar keine Verschlüsselung im engeren Sinne, sondern eine Verschleierung der Information.

Die Kryptographie ist die Wissenschaft der Verschlüsselung. Sie ist ein wichtiges Gebiet der Informatik. Wirtschaftsunternehmen und Banken, die das Internet nutzen, sind auf sichere Verschlüsselung angewiesen. Moderne Verschlüsselungen werden mit dem Computer durchgeführt und beruhen auf mathematischen Verfahren, die es praktisch unmöglich machen eine geheime Nachricht zu lesen, wenn man nicht den Schlüssel kennt.

Webseiten und Stichwörter

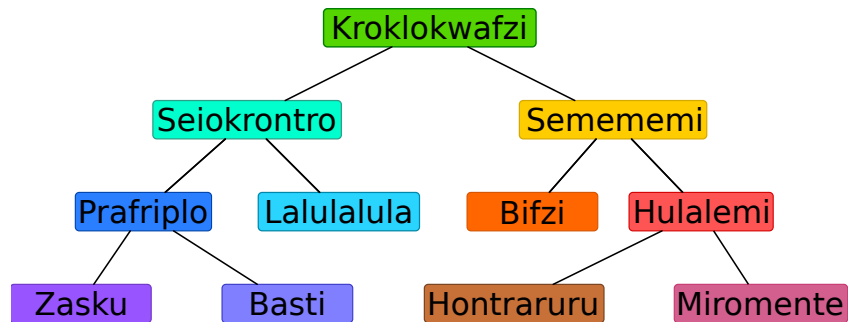
Verschlüsselung, Kryptographie, Transposition, Datenschutz

- [https://de.wikipedia.org/wiki/Transposition_\(Kryptographie\)](https://de.wikipedia.org/wiki/Transposition_(Kryptographie))



2. Hierarchie

Das Bild beschreibt die Beziehungen zwischen Tierarten auf dem Planet Morgenstern. Eine Linie zwischen zwei Tierarten bedeutet, dass alle Tiere der unteren Art auch Tiere der oberen Art sind.



Beispielsweise sind alle „Hulalemi“ auch „Semememi“. Manche „Seiokronro“ sind hingegen keine „Basti“.

Nur eine der folgenden Behauptungen ist richtig. Welche?

- A) Alle Basti sind auch Seiokronro.
- B) Manche Hontraruru sind keine Semememi.
- C) Alle Zasku sind auch Bifzi.
- D) Alle Prafriplo sind auch Basti.



Lösung

Die richtige Antwort ist A).

A): Alle Tiere der Art Basti sind auch Tiere der Art Prafriplo. Da alle Prafriplos Tiere der Art Seiokrontro sind, sind auch alle Bastis Tiere der Kategorie Seiokrontro.

B): Hontrarurus sind Tiere der Art Hulalemi, welche wiederum Tiere der Art Semememi sind. Deshalb sind alle Hontrarurus Tiere der Art Semememi.

C): Zaskus sind Tiere der Art Prafriplo, nicht jedoch der Art Bifzi.

D): Alle Bastis sind Tiere der Art Prafriplo, nicht jedoch umgekehrt.

Dies ist Informatik!

Die Beziehungen der Tierarten sind in einer Form notiert, die in der Informatik „Baum“ genannt werden. Biologen nutzen dies als „Phylogenetischer Baum“, der Beziehungen zwischen Tierarten darstellen.

In der Informatik werden Bäume oft verwendet um Beziehungen graphisch darzustellen. So zeigt ein Familienstammbaum beispielsweise die Beziehung zwischen Kindern, Eltern und Großeltern. Wenn Beziehungen in Form von Bäumen dargestellt werden, kann schnell festgestellt werden, welche Beziehung zwischen zwei Knotenpunkten besteht.

Bäume eignen sich darüber hinaus hervorragend, um Elemente geordnet abzuspeichern, bzw. um Elemente rasch zu finden ... man benötigt nur wenige Schritte, um riesige Mengen von Daten zu durchsuchen.

Webseiten und Stichwörter

Bäume, Spezialisierung, Generalisierung

- [https://de.wikipedia.org/wiki/Baum_\(Graphentheorie\)](https://de.wikipedia.org/wiki/Baum_(Graphentheorie))
- https://de.wikipedia.org/wiki/Phylogenetischer_Baum



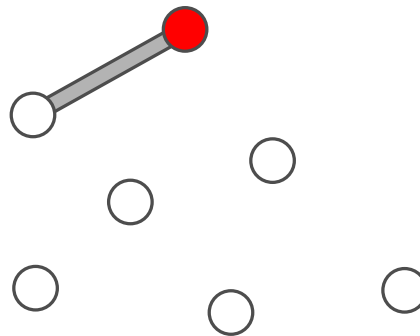
3. Brückenbau

Der Biber-Opa ist ein wenig wasserscheu geworden. Er möchte von seiner Burg zu allen anderen Burgen der Biber-Familie über Brücken gehen können. Die Biber meinen es gut mit Opa und wollen beim Brückenbauen folgendes beachten:

- Opa soll von seiner Burg aus höchstens über zwei Brücken gehen müssen.
- Neben der Brücke, mit der man zu einer Burg kommt, dürfen höchstens zwei weitere Brücken davon wegführen.

Die Biber beginnen mit einem Brückenplan. Sie zeichnen alle Burgen als Kreise. Opas Burg ist ein rot ausgefüllter Kreis. Eine erste Brücke von Opas Burg aus zeichnen sie ein. Aber dann wissen sie nicht mehr weiter.

Vervollständige den Plan so, dass er alle Bedingungen erfüllt. Dazu gibt es mehrere Möglichkeiten. Auf jeden Fall werden fünf weitere Brücken benötigt.

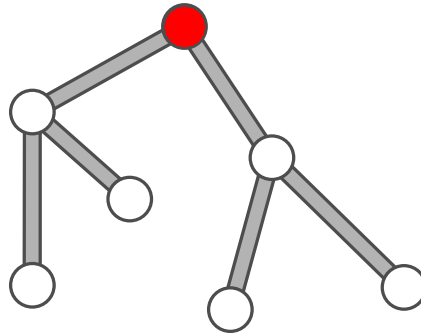




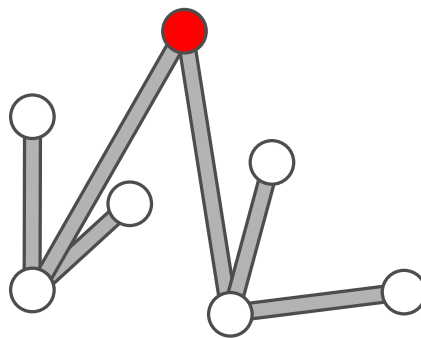
Lösung

Der folgende Plan erfüllt alle Bedingungen, denn:

- vom roten Kreis aus ist jeder andere Kreis über höchstens zwei Linien zu erreichen
- von jedem Kreis gehen höchstens drei Linien aus.



Es gibt auch noch andere Pläne, die richtig sind, zum Beispiel:



Dies ist Informatik!

Obwohl es viele mögliche Lösungen gibt, haben sie alle denselben Aufbau. Von der roten Burg aus gehen zwei Brücken zu zwei anderen Burgen und von diesen beiden anderen Burgen aus gehen je zwei Brücken zu den verbleibenden vier Burgen. Gäbe es eine Burg mehr, könnte man diese nicht unter Einhalten der Regeln auch noch erreichen.

Die Biber bauen hier einen sogenannten *Baum* auf: alle *Knoten* (die Burgen) können erreicht werden, indem man *Kanten* (die Brücken) entlang läuft. Die beiden Regeln stellen sicher, dass ein besonderer Baum erzeugt wird: die Tatsache, dass nur zwei Kanten weiterführen dürfen, erzeugen einen *Binärbaum*. Die Tatsache, dass man maximal zwei Kanten bis zu jedem Knoten entlang laufen darf, stellt sicher, dass der Baum *minimal* ist.

Webseiten und Stichwörter

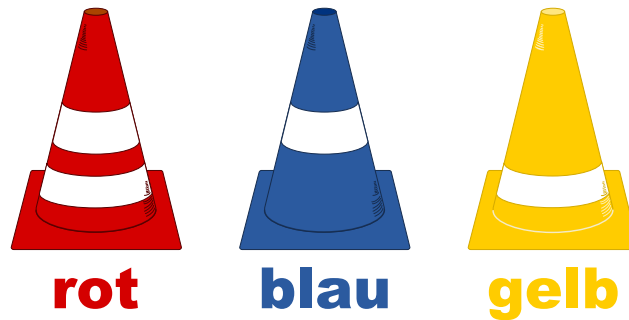
Baum, Burgen

- [https://de.wikipedia.org/wiki/Baum_\(Graphentheorie\)](https://de.wikipedia.org/wiki/Baum_(Graphentheorie))
- <https://de.wikipedia.org/wiki/Binärbaum>



4. Verkehrshüetli-Versteck

Vreni versteckt Karten unter drei Verkehrskegeln. Dabei legt sie immer eine Karte unter einen bestimmten Kegel. Wenn dort schon eine Karte liegt, legt sie die neue Karte oben auf die alte Karte.



Welche Karte sie unter welchem Verkehrskegel versteckt hat, schreibt sie sich so auf:

rot + 5

rot + 3



Vreni hat sich notiert:

rot + 3

gelb + 5

rot + 6

gelb + 8

blau + 1

gelb + 3

Welche Karte liegt am Ende unter den jeweiligen Verkehrskegel zuoberst?

- A) rot: 3, blau: 1, gelb: 5
- B) rot: 9, blau: 1, gelb: 16
- C) rot: 6, blau: 1, gelb: 3
- D) rot: 8, blau: 1, gelb: 3



Lösung

Am Anfang liegt unter keinem Verkehrskegel eine Karte. Die folgende Tabelle zeigt an, welche Karten jeweils nach jedem Schritt unter den Verkehrskegeln liegen:

Befehl	rot	blau	gelb
	-	-	-
rot ← 5	5	-	-
gelb ← 5	5	-	5
rot ← 6	6	-	5
gelb ← 8	6	-	8
blau ← 1	6	1	8
gelb ← 3	6	1	3

Damit ist klar, dass die Antwort C) die richtige Lösung ist.

Dies ist Informatik!

Ein Verkehrskegel kann als eine Variable aufgefasst werden, deren Name „rot“, „blau“ oder „gelb“ ist. Eine Variable ist ein Ort im Speicher des Computers, an dem sich der Computer Werte merken kann, in diesem Fall sind es ganze Zahlen. Jeder Befehl ist eine Zuweisung, das heisst dass der bisherige Wert der gewählten Variablen durch einen neuen Wert überschrieben wird.

Zuweisungen werden in der Informatik häufig mit „:=“ oder „←“ notiert. Das führt jedoch häufig zu Denkfehlern, denn während in der Mathematik der Ausdruck „ $x = x + 1$ “ falsch ist, bedeutet er in der Informatik, dass man den Wert der Variablen x nimmt, ihn um 1 erhöht, und das Ergebnis als neuen Wert in der Variablen x speichert. Daher wird ausserhalb von Programmiersprachen oftmals der Pfeil „←“ für eine Zuweisung verwendet, denn „ $x ← x + 1$ “ ist nun auch für Mathematiker verständlich, während Informatiker sich nicht gross dran stören.

Webseiten und Stichwörter

Variable, Zuweisung

- [https://de.wikipedia.org/wiki/Variable_\(Programmierung\)](https://de.wikipedia.org/wiki/Variable_(Programmierung))
- <https://de.wikipedia.org/wiki/Zuweisung>

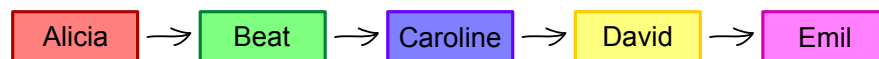


5. Gruppenarbeit

Sarah und ihre Mitschüler Alicia, Beat, Caroline, David und Emil sollen für den Unterricht eine Gruppenarbeit machen. Sie teilen sich die Arbeit auf, wobei Sarah die Aufgabe hat, die Ergebnisse zu sammeln. Das Ergebnis von Emil kann sie sich direkt geben lassen, aber um von den anderen die Ergebnisse zu bekommen, muss sie zunächst andere Ergebnisse vorzeigen:

- Um von David das Ergebnis zu bekommen, muss sie das Ergebnis von Alicia vorzeigen.
- Um von Beat das Ergebnis zu bekommen, muss sie das Ergebnis von Emil vorzeigen.
- Um von Caroline das Ergebnis zu bekommen, muss sie die Ergebnisse von Beat und David vorzeigen.
- Um von Alicia das Ergebnis zu bekommen, muss sie die Ergebnisse von Beat und Emil vorzeigen.

Ziehe die Namen in eine Reihenfolge, bei der Sarah alle Ergebnisse erhalten kann:





Lösung

Die richtige Lösung ist:

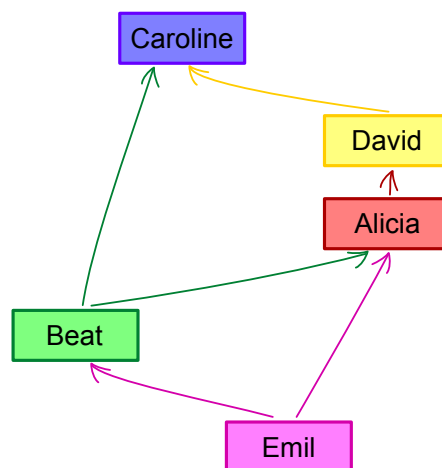


Emil ist der Einzige, der sein Ergebnis herausgibt, ohne dass er die Ergebnisse von den anderen sehen will, also muss er als erstes befragt werden. Beat ist der einzig übrige, der sein Ergebnis herausgibt, wenn ihm nur das Ergebnis von Emil gezeigt wird. Alicia ist dann die einzig übrige, die ihr Ergebnis herausgibt, wenn ihr nur die Ergebnisse von Emil und Beat gezeigt werden. David ist wiederum der einzig übrige, der sein Ergebnis herausgibt, wenn ihm nur die Ergebnisse von Emil, Beat und Alicia gezeigt werden (ihn interessiert dabei lediglich das Ergebnis von Alicia). Übrig bleibt Caroline, da man ihr aber die Ergebnisse von Beat und David zeigen kann, gibt auch sie ihr Ergebnis heraus.

Dies ist Informatik!

Es passiert im Leben häufig, dass man bestimmte Dinge erst machen kann, wenn man bestimmte Voraussetzungen dafür erfüllt hat. Die Informatik beschäftigt sich mit solchen Ablaufplanungen (Scheduling) ebenfalls. Die Programme die hierfür entwickelt werden, werden beispielsweise in der Industrie eingesetzt. Die komplexen Abläufe wie den Bau eines Autos können so möglichst effizient, also zum Beispiel ohne unnötiges Warten auf Teile, gestaltet werden. Damit spart man viel Geld, weil man kein grosses Lager mehr benötigt und Maschinen möglichst viel arbeiten können.

Um einen Ablauf zu modellieren, erstellt man einen Graph, dessen Knoten die einzelnen Vorgänge sind (in diesem Fall das Abholen von Ergebnissen bei einer Person), und dessen Kanten durch Pfeile (sogenannte gerichtete Kanten, so dass ein gerichteter Graph entsteht) die Voraussetzungen darstellen. Ein solcher Graph für unser Problem sieht so aus:



Eine mögliche Lösung eines Ablaufplans ist ein Pfad durch den Graphen, der alle Knoten besucht. In unserem Graphen gibt es nur einen möglichen Pfad, nämlich die korrekte Lösung Emil → Beat → Alicia → David → Caroline.

Wie oben angedeutet, ist das Finden einer Lösung jedoch nur ein Teil des Problemlösens. Es gibt komplexere Situationen, wenn beispielsweise Dinge parallel stattfinden können. Zudem ist es häufig wichtig, eine Lösung zu finden, die beispielsweise die Kosten möglichst gering wählt, oder die besonders schnell sind. Oftmals jedoch ist es nicht praktikabel, einfach alle Lösungen auszuprobieren, so dass man sich mit Algorithmen zur Annäherung einer optimalen Lösung behilft.



Webseiten und Stichwörter

Scheduling, Ablaufplan, gerichteter Graph

- <https://de.wikipedia.org/wiki/Scheduling>



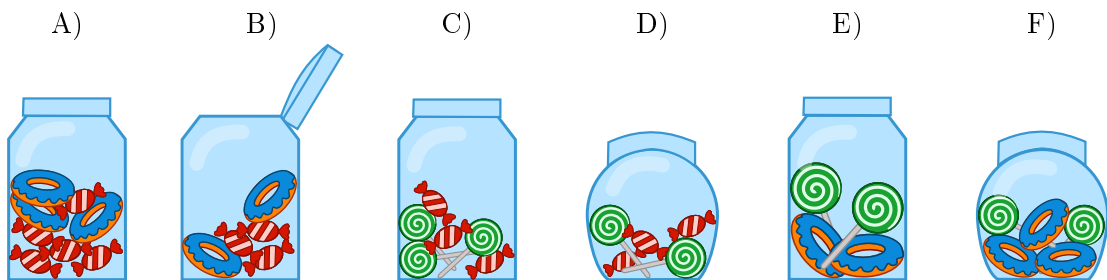


6. Bonbonnieren

Eine Bonbonniere ist ein Glasgefäß für Süßigkeiten. Carl und Judy haben je drei Bonbonnieren. Carl gehören die Bonbonnieren A), B) und C), während Judy die Bonbonnieren D), E) und F) gehören. Jede Bonbonniere hat folgende fünf Eigenschaften:

- Die Bonbonniere ist entweder offen oder geschlossen.
- Die Bonbonniere enthält rot-weiss gestreifte Bonbons oder nicht.
- Die Bonbonniere enthält blaue Zuckerringe oder nicht.
- Die Bonbonniere enthält Lutscher mit grünen Spiralen oder nicht.
- Die Bonbonniere ist rund oder eckig.

Wähle die Bonbonniere aus, die sowohl die gemeinsamen Eigenschaften von Carls Bonbonnieren als auch die gemeinsamen Eigenschaften von Judys Bonbonnieren hat.





Lösung

Die blauen Zuckerringe spielen keine Rolle, weil sie weder von Carls als auch von Judys Bonbonnieren eine gemeinsame Eigenschaft sind. Sie fehlen in C) und D).

Die zwei gemeinsamen Eigenschaften von Carls Bonbonnieren A), B) und C) sind:

- jede Bonbonniere ist eckig;
- in jeder Bonbonniere sind rot-weiss gestreifte Bonbons.

Die zwei gemeinsamen Eigenschaften von Judys Bonbonnieren D), E) und F) sind:

- jede Bonbonniere ist geschlossen;
- in jeder Bonbonniere sind Lutscher mit grünen Spiralen.

Nur die Bonbonniere C) hat alle vier Eigenschaften: Sie ist eckig, es sind rot-weiss gestreifte Bonbons und Lutscher mit grünen Spiralen darin, und sie ist geschlossen.

Dies ist Informatik!

Die Informatik fasst bei ihren Datenmodellen gern Objekte aufgrund von Eigenschaften in Gruppen zusammen. In dieser Biberaufgabe haben wir es mit fünf Eigenschaften und zwei Gruppen zu tun. Wir suchen nach Objekten, welche die gemeinsamen Eigenschaften beider Gruppen haben. Bei relationalen Datenbanken nennt man das auch „den Durchschnitt zweier Mengen bilden“.

Bei vielen Datenbanken kann man gesuchte Objekte anhand gewünschter Eigenschaften aus grossen unübersichtlichen Objektmengen herausfiltern („eine Teilmenge bilden“). Zum Beispiel kann man auf diese Weise bei einem Online-Shop gezielt nach Smartphones mit einer bestimmten Akkulaufzeit, Displaygrösse und anderen in der Datenbank abgespeicherten Eigenschaften suchen.

Beim Aufbau einer Datenbank ist sehr darauf zu achten, welche Eigenschaften im Datenmodell vorkommen. Werden wichtige Eigenschaften vergessen, ist alles spätere Suchen weniger treffsicher. Werden überflüssige Eigenschaften mit modelliert, macht das die spätere Dateneingabe teurer, ohne das sich der Nutzen der Datenbank erhöht.

Webseiten und Stichwörter

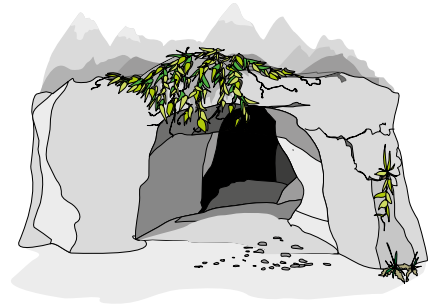
Objektorientierung, Attribute, Logik

- [https://de.wikipedia.org/wiki/Menge_\(Mathematik\)#Durchschnitt_.28Schnittmenge.2C_Schnitt.29](https://de.wikipedia.org/wiki/Menge_(Mathematik)#Durchschnitt_.28Schnittmenge.2C_Schnitt.29)
- [https://de.wikipedia.org/wiki/Menge_\(Mathematik\)#Vereinigung_.28Vereinigungsmenge.29](https://de.wikipedia.org/wiki/Menge_(Mathematik)#Vereinigung_.28Vereinigungsmenge.29)
- <https://de.wikipedia.org/wiki/Mengenlehre>
- <https://de.wikipedia.org/wiki/Datenbank>



7. Durch den Tunnel

Anna und Benno machen mit ihren Eltern eine Wanderung. Auf ihrer Strecke liegt ein Tunnel. Aus Erfahrung wissen sie, dass jeder von ihnen unterschiedlich viel Zeit für die Tunnelpassage benötigt: Anna benötigt 10 Minuten, Benno 5 Minuten, die Mutter 20 Minuten und der Vater 25 Minuten.



Den dunklen und engen Tunnel kann man nur alleine oder zu zweit passieren. Sie müssen also mehrere Passagen machen. Zu zweit benötigt man so viel Zeit wie die langsamere der beiden Personen. Im Tunnel muss man auf jeden Fall eine Lampe benutzen.

Als sie an den Eingang des Tunnels kommen, stellen sie fest: Der Akku ihrer einzigen Lampe reicht nur noch für 60 Minuten. Können sie innerhalb dieser 60 Minuten alle durch den Tunnel kommen? Anna behauptet: „Ja, können wir, und zwar mit fünf Passagen!“

Ziehe die Namen so in die passenden Felder, dass Annas Plan umgesetzt wird.

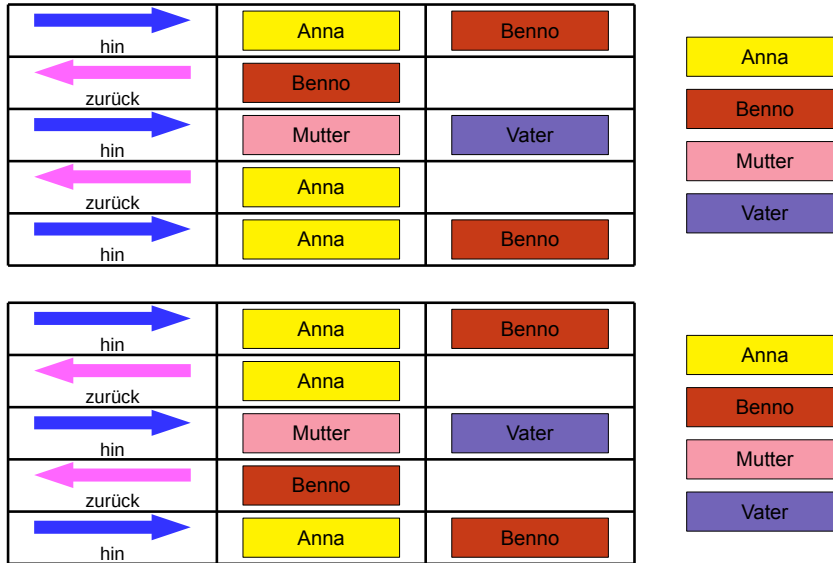
→ hin		
← zurück		
→ hin		
← zurück		
→ hin		

Anna
Benno
Mutter
Vater



Lösung

Dass alle innerhalb von 60 Minuten am Ausgang sind, ist nur möglich, wenn Mutter und Vater (die beiden langsamsten) nur einmal den Tunnel passieren. Das wiederum ist nur möglich, wenn vorher Anna und Benno den Tunnel passieren und einer von ihnen zurück geht. Der eine kann dann die Lampe Vater und Mutter geben, und der andere bringt sie durch den Tunnel zurück, damit dann beide rechtzeitig zusammen zum Ausgang gehen können. Dabei ist es egal, ob in Passage 2 Anna zurückgeht oder Benno: In Passage 4 ist der jeweils andere dran, so dass diese beiden Passagen in beiden Fällen zusammen 15 Minuten dauern.



Dies ist Informatik!

In der Informatik kommt es häufig vor, dass man nur wenige *Ressourcen* hat, um ein Problem zu lösen. Gleichzeitig erfordert das Lösen eines Problems aber auch das Erfüllen bestimmter *Nebenbedingungen*. Das Problem wird gelöst, indem man einen Plan aufstellt, der die vorgegebenen Ressourcen nicht überschreitet und alle Nebenbedingungen erfüllt.

In der Informatik werden solche Probleme mit Hilfe von *Scheduling-Algorithmen* gelöst. Da Probleme aus der Praxis häufig so kompliziert sind, dass ein Computer lange (möglicherweise sogar mehrere Jahre) rechnen müsste, um eine optimale Lösung zu finden, begnügt man sich dann damit, eine mögliche Lösung zu finden, die gut genug ist. Das reicht in diesem Fall auch: es ist nicht danach gefragt, den schnellsten Weg durch den Tunnel zu finden, sondern einen Weg, der nicht länger als 60 Minuten dauert. Dass es zufälligerweise egal ist, dass es keinen schnelleren Weg gibt, stört nicht, da man ja nach dem Finden einer Lösung aufhören kann zu suchen.

Webseiten und Stichwörter

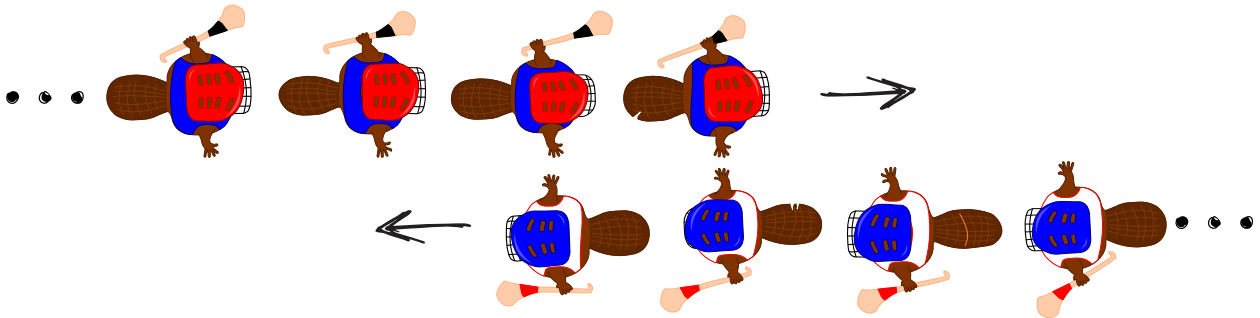
Scheduling, Nebenbedingungen, Ressourcen

- <https://de.wikipedia.org/wiki/Scheduling>



8. Hände schütteln

Biber spielen gerne das irische Spiel Hurling. Am Schluss einer Partie Hurling stellen sich beide Mannschaften hintereinander in einer Reihe auf. Dann laufen die Spieler aneinander vorbei, schütteln sich nach und nach die Hände und sagen „Danke für das Spiel!“



Das Händeschütteln läuft im einzelnen so ab: Zuerst schütteln sich die beiden ersten Spieler die Hände. Dann schütteln die ersten Spieler den zweiten Spielern der jeweils anderen Mannschaft die Hände (siehe Bild). Dies geht so weiter, bis auch die beiden letzten Spieler sich die Hände geschüttelt haben.

Beim Hurling gibt es 15 Spieler pro Mannschaft. Dass zwei Spieler sich die Hände schütteln und zum jeweils nächsten Spieler gehen, dauert 1 Sekunde.

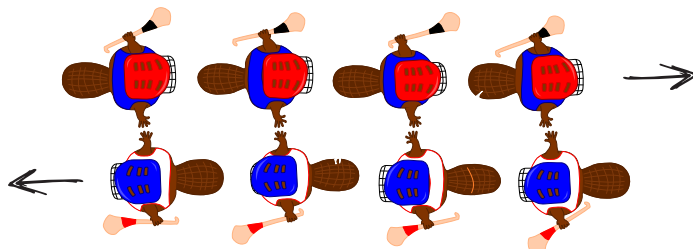
Wie viele Sekunden dauert das Händeschütteln der beiden Mannschaften insgesamt?



Lösung

29 ist die richtige Antwort.

Das Händeschütteln geht so: Der jeweils erste Spieler jeder Mannschaft schüttelt jedem Spieler der anderen Mannschaft die Hand. Bei vier Spielern wären das vier Sekunden und die Situation wäre die folgende:



Gleichzeitig schüttelt der jeweils letzte Spieler jeder Mannschaft zuerst einem Spieler der anderen Mannschaft die Hand. Er muss aber noch allen anderen Spielern die Hand schütteln, was bei vier Spielern weitere drei Sekunden wären.

Allgemein dauert es bei n Spielern also zuerst n Sekunden und danach noch einmal $n - 1$ Sekunden, so dass insgesamt $n + n - 1 = 2n - 1$ Sekunden sind. Bei 15 Spielern sind es also $15 + 15 - 1 = 29$ Sekunden, die das Händeschütteln dauert.

Dies ist Informatik!

Für Hurling-Teams mit 15 Spielern konnten wir genau berechnen, wie lange das Händeschütteln braucht. 29 Sekunden sind für die Zuschauer gut auszuhalten. Doch wie sieht diese „Laufzeit“ bei Eishockey-Teams mit insgesamt 22 Spielern aus? Ist der Algorithmus des Hurling-Händeschütteln dann immer noch brauchbar, oder würde es zu lange dauern? Es wäre gut, eine allgemeine Einschätzung der Laufzeit eines Algorithmus zu haben, ohne dass wir immer alles einzeln berechnen müssen.

Die Informatik befasst sich intensiv mit allgemeinen Einschätzungen von Algorithmen-Laufzeit. Solche Laufzeitanalysen liefern einen mathematischen Ausdruck, der eine Variable n für die Grösse der Eingabe enthält. Für das Hurling-Händeschütteln erhalten wir einen solchen Ausdruck, wenn wir im zweiten Satz der Answererklärung „Anzahl der Spieler einer Mannschaft“ durch n ersetzen: $2n - 1$. Damit lässt sich auch für andere Spielerzahlen die Händeschüttel-Laufzeit exakt berechnen: für 22 Spieler 43 Sekunden, für 40 Spieler 79 Sekunden usw.

Hinter dem Laufzeitausdruck $2n - 1$ steckt eine lineare Funktion. Damit gehört der Händeschüttel-Algorithmus zur Klasse der Algorithmen mit linearer Laufzeit, die man auch als $O(n)$ bezeichnet. Wie wäre es aber, wenn man anders die Hände schütteln würde? Beispielsweise dass jeder jedem einzeln die Hand gibt? Dann gehörte er zur Klasse $O(n^2)$, die Hurling-Teams würden dann $15^2 = 225$ Sekunden lang Hände schütteln, also fast vier Minuten. Hätte der Algorithmus gar exponentielle Laufzeit, wäre also in der Klasse $O(2^n)$, würde ca. $2^{15} = 32768$ Sekunden lang Hände geschüttelt, also über 9 Stunden. Da wären die Zuschauer längst im Bett. Es lohnt sich also, gut darüber nachzudenken, ob man nicht wie in diesem Fall einige Dinge parallel erledigen kann, um Zeit zu sparen.

Webseiten und Stichwörter

Laufzeitkomplexität, Laufzeitanalyse



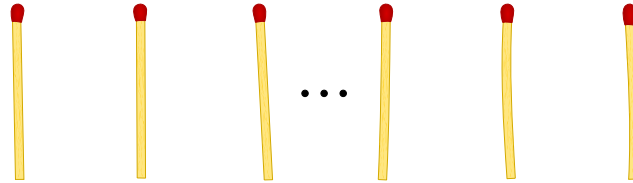
- [https://de.wikipedia.org/wiki/Laufzeit_\(Informatik\)](https://de.wikipedia.org/wiki/Laufzeit_(Informatik))





9. Nim

Beat und sein Freund spielen das Nim-Spiel. 13 Hölzchen liegen auf dem Tisch. Die beiden Spieler nehmen abwechselnd 1, 2 oder 3 Hölzchen weg. Wer das letzte Hölzchen nimmt, hat gewonnen.



Hinweis: Wenn noch vier Hölzchen auf dem Tisch liegen, kann Beat nicht mehr gewinnen. Diese Situation möchte er vermeiden.

Beat fängt an. Wie viele Hölzchen muss er wegnehmen, um das Spiel zu gewinnen?

- A) 1
- B) 2
- C) 3
- D) Das spielt keine Rolle.



Lösung

1 ist richtig. Dann bleiben 12 Hölzchen übrig. Der Freund nimmt 1, 2 oder 3 weg und Beat nimmt so viele, dass 8 übrig bleiben. Wieder nimmt der Freund 1, 2 oder 3 weg. Beat nimmt so viele, dass 4 übrig bleiben und der Freund nicht mehr gewinnen kann.

Wenn Beat 2 oder 3 Hölzchen nimmt, kann der Freund so reagieren, dass ein Vielfaches von 4 übrig bleibt. Dann kann Beat nicht mehr gewinnen.

Dies ist Informatik!

In der Spieltheorie werden Spiele wie das Nim-Spiel oder das berühmte Gefangenens-Dilemma als Modelle verwendet, um strategische Probleme in der Wirklichkeit zu analysieren und Lösungswege zu finden. In einer Marktwirtschaft dienen die Erkenntnisse zum Beispiel dazu, Preise optimal zu gestalten. Eine Preissenkung kann zwar den Absatz erhöhen, aber sie verringert gleichzeitig den Gewinn pro verkauftem Produkt. Umgekehrt vergrößert eine Preissteigerung den Gewinn pro verkauftem Produkt, aber sie kann auch dazu führen, dass der Absatz und damit der Gesamtgewinn sinkt. Mit Modellen aus der Spieltheorie können mögliche Reaktionen der Käufer auf Preisänderungen vorhergesagt werden. Wie bedeutend diese Modelle für die Wirtschaft sind, sieht man daran, dass bereits mehrere Nobelpreise für spieltheoretische Arbeiten verliehen wurden.

Webseiten und Stichwörter

Nim-Spiel, Spieltheorie, Entscheidungsbaum

- <http://www.mathematische-basteleien.de/nimspiel.html>
- <http://scienceblogs.de/zoonpolitikon/2008/04/22/spieltheorie-einfach-erklart-i-einleitung-und-gefangenendilemma/>



10. Sortierte Trikot-Nummern

Die folgenden Bilder zeigen zwei Teams mit je 15 Spielern. Beide Teams haben mit Nummern bedruckte Trikots. Die Spieler der ersten Mannschaft sind nach ihrer Nummer sortiert. Die Spieler der zweiten Mannschaft sind nicht sortiert.

Team 1:



Team 2:



Wie kann man am schnellsten feststellen, welche Nummern sowohl im Team 1 als auch im Team 2 verwendet werden?

- A) Man durchläuft die Nummern von Team 1 (1, 4, 5, ...). Bei jeder Nummer stellt man fest, ob diese in Team 2 vorkommt.
- B) Man durchläuft die Nummern von Team 2 (8, 28, 12, ...). Bei jeder Nummer stellt man fest, ob diese in Team 1 vorkommt.
- C) Es ist egal bei welchem Team man beginnt. Beides geht gleich schnell.
- D) Zuerst muss man feststellen, wie viele Nummern *nicht* in beiden Teams vorkommen. Ziehe ich diese Zahl von 15 ab, dann erhalte ich die gesuchte Zahl.



Lösung

Die richtige Antwort ist B). In einer sortierten Folge findet man Zahlen schneller als in einer nicht sortierten Folge. Ein Computer würde beispielsweise nur $\log_2(n)$ anstelle von n Suchschritten benötigen. Somit benötigt Antwort B) im dümmsten Fall $n \cdot \log_2(n)$ Schritte.

Für die Antwort A) würde man zum Durchsuchen der sortierten Folge nicht schnell suchen, sondern müsste alle Zahlen einzeln durchgehen. Somit benötigt Antwort A) im dümmsten Fall n^2 Schritte, was mehr als $n \cdot \log_2(n)$ Schritte ist.

Damit ist auch widerlegt, dass die Antwort C) richtig wäre, denn n^2 ist nicht gleich $n \cdot \log_2(n)$.

Die Antwort D) gibt in der Regel nicht die richtige Antwort, daher kann sie also schon gar nicht der schnellste Weg zur richtigen Antwort sein.

Dies ist Informatik!

Eine sortierte Folge kannst du schneller durchsuchen als eine unsortierte Folge. Wie suchst du z.B. die Nummer 9 auf den Trikots von Team 1? Du schaust zuerst auf den mittleren Spieler. Er hat die Nummer 17. Die gesuchte Nummer 9 ist kleiner. Also suchst du links weiter und guckst auf die Mitte der linken Hälfte der Mannschaft. Da steht Nummer 7. Also suchst du rechts von 7 weiter und hast die Nummer 9 rasch gefunden. Der Trick ist, dass du das Suchgebiet bei jedem Suchschritt in zwei Teile teilst und damit immer kleiner machst. Dieses Verfahren nennt man in der Informatik binäre Suche. Das kommt vom lateinischen Wort „bis“, was übersetzt „zwei Mal“ bedeutet. Weil sortierte Sammlungen von Daten durch binäre Suche schnell durchsucht werden können, ist das Sortieren in der Informatik eine wichtige Sache. Wenn n Zahlen durchsucht werden, sind das dann nur noch ungefähr $\log_2(n)$ Suchschritte.

Webseiten und Stichwörter

Sortierung, binäre Suche

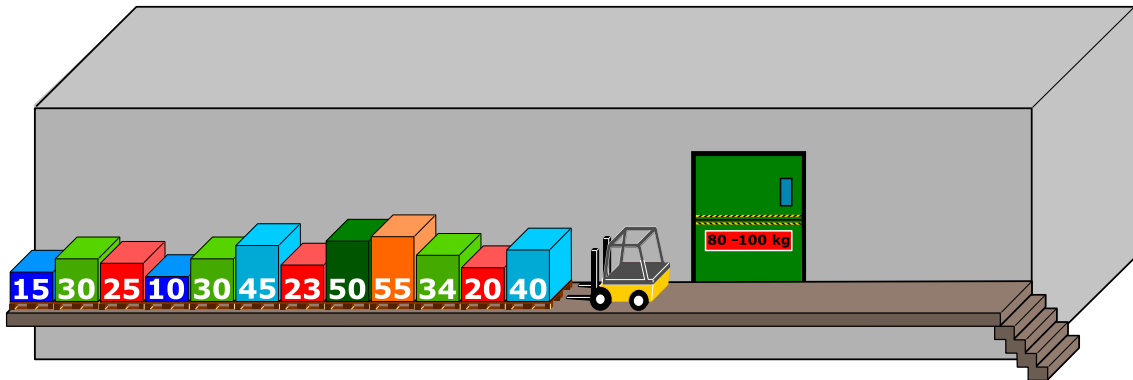
- https://de.wikipedia.org/wiki/Binäre_Suche



11. Palettenlift

Auf einer engen Laderampe stehen Paletten nebeneinander. Die Paletten wiegen (von links nach rechts): 15 kg, 30 kg, 25 kg, 10 kg, 30 kg, 45 kg, 23 kg, 50 kg, 55 kg, 34 kg, 20 kg, 40 kg. Die Rampe ist so eng, dass die Paletten nicht aneinander vorbei gehoben werden können.

Die Paletten werden in einen Lift geladen, der sie zum Lager bringt. Der Lift bringt die Paletten weg, sobald er mit mindestens 80 kg Ware geladen ist. Es darf jedoch nicht mehr als 100 kg wiegen. Er kommt dann wieder leer zurück.



Beim Laden der Paletten in den Lift wird immer diejenige Palette genommen, die sich am nächsten zum Lift befindet. Wenn das Gesamtgewicht des Liftes beim Laden von der letzten Palette 100 kg übersteigt, wird die Palette auf die andere Seite der Rampe gebracht. Ansonsten bleibt die Palette im Lift.

Wenn alle Paletten von links zum Lager gebracht worden sind, werden die auf der anderen Seite der Rampe in derselben Art und Weise in den Lift geladen.

Welche der folgenden Aussagen ist korrekt?

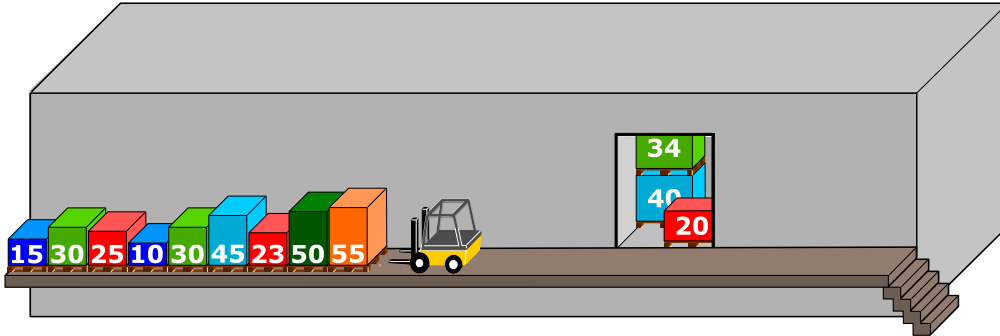
- A) Bei der zweiten Fahrt des Lifts wiegt seine Ladung 98 kg.
- B) Die Paletten am anderen Ende der Rampe werden nicht verwendet.
- C) Bei einer Fahrt des Lifts wiegt seine Ladung 100 kg.
- D) Der Lift macht insgesamt fünf Fahrten.
- E) Es ist nicht möglich, die Paletten nach der oben genannten Prozedur zum Lager zu bringen.



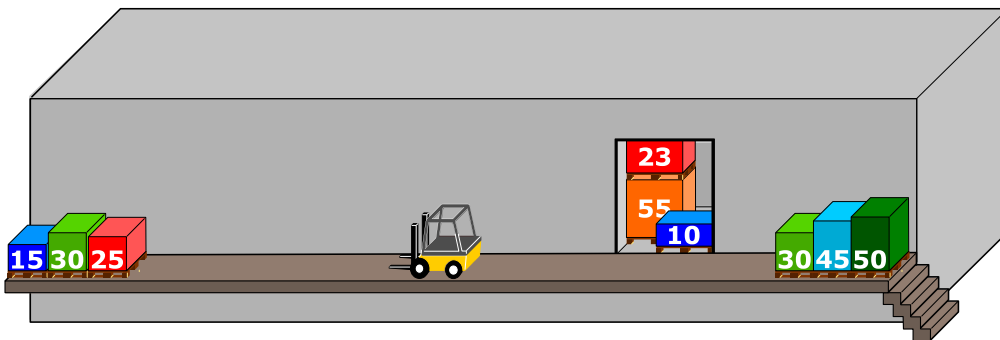
Lösung

Die korrekte Antwort ist C):

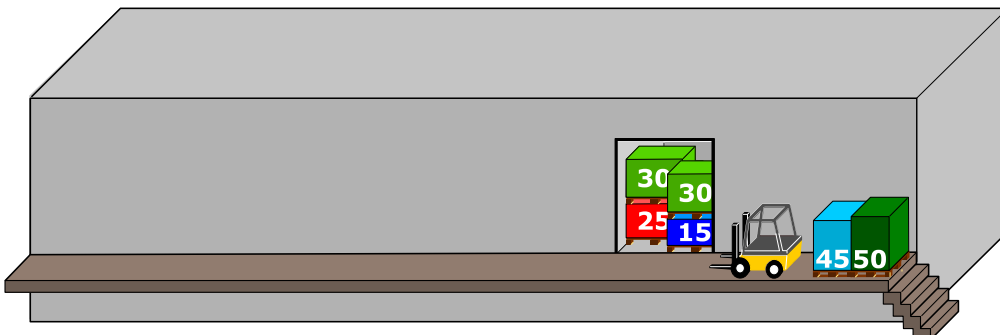
- Die erste Ladung besteht aus den ersten drei Paletten. Diese wiegen $40\text{ kg} + 20\text{ kg} + 34\text{ kg} = 94\text{ kg}$.



- Die zweite Ladung besteht zunächst aus einer einzigen Palette (55 kg). Wenn die nächste Palette (50 kg) ebenfalls im Lift geladen werden würde, würde die Ladung die 100 kg übersteigen, deswegen wird diese Palette zum anderen Ende der Rampe gebracht. Die nächste Palette (23 kg) wird in den Lift gestellt; jedoch ist die Ladung nun zu leicht mit $55\text{ kg} + 23\text{ kg} = 78\text{ kg}$. Die nächste Palette der Reihe (45 kg) würde wieder das zugelassene Gewicht im Lift übersteigen. Aus diesem Grund wird es ebenfalls auf der anderen Seite der Rampe gebracht. Dasselbe geschieht mit der nächsten Palette (30 kg). Letztendlich kann die nächste Palette (10 kg) in den Lift gestellt werden. Der Lift wird zum Laden gefahren mit einer Ladung von $55\text{ kg} + 23\text{ kg} + 10\text{ kg} = 88\text{ kg}$.

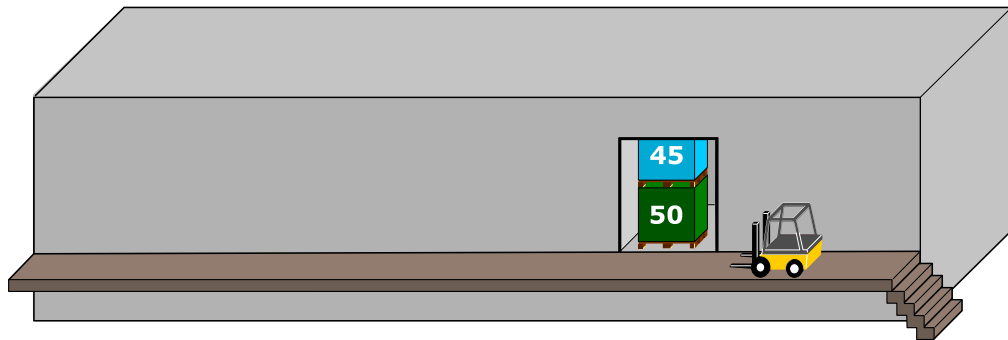


- Die letzten drei Paletten der Reihe ($25\text{ kg} + 30\text{ kg} + 15\text{ kg} = 70\text{ kg}$) werden ebenfalls in den Lift geladen. Der dritten Ladung wird noch die erste (30 kg) der Paletten in der gegenüberliegenden Reihe hinzugefügt, so dass die Ladung im Lift genau 100 kg wiegt.





- Zu diesem Zeitpunkt sind nur noch zwei Paletten auf der anderen Seite der Rampe übrig: 45 kg + 50 kg = 95 kg. Diese werden als vierte Ladung im Lift zum Laden gefahren.



Ausgehend von dieser Erklärung sind die Antworten A), B), D) und E) nicht korrekt.

Dies ist Informatik!

Weil die Rampe so eng ist, kann immer nur die erste Palette der Reihe bearbeitet werden. Das ist wie bei einem Stapel: Nur das oberste Ding auf dem Stapel kann genommen werden, und neue Dinge können nur oben auf den Stapel gelegt werden. In der Informatik heisst diese Datenstruktur auch konsequenterweise „Stapel“. In diesem Fall kommen drei Stapel vor: die linke Reihe von Paletten (mit dem obersten Element ganz rechts), die (anfangs leere) rechte Reihe von Paletten (mit dem obersten Element ganz links) und der Lift.

Stapel kommen auch sonst häufig vor: wenn ein Kind einen Turm aus Bauklötzen baut, kann es in der Regel nur den obersten Bauklotz entfernen oder oben einen drauflegen. Eine Glace mit mehreren Kugeln verhält sich meist ebenfalls wie ein Stapel: die Kugel, die zuerst auf dem Cornet platziert wird, wird zuletzt gegessen. Beim Surfen im Web können die besuchten Seiten im Tab ebenfalls als Stapel betrachtet werden: wenn auf „Zurück“ geklickt wird, erscheint die zuletzt besuchte Seite als erste.

Das Prinzip dahinter nennt man auch LIFO: „last in – first out“ (im Gegensatz zu einer Schlange, die nach dem FIFO-Prinzip funktioniert: „first in – first out“). Dieses Prinzip wird beispielsweise bei einer Warteschlange am Schalter oder beim Arzt verwendet (wobei Ärzte zum Glück Notfälle vorziehen können).

Webseiten und Stichwörter

Datenstruktur, Stapel, Algorithmus, LIFO-Prinzip

- <https://de.wikipedia.org/wiki/Stapelspeicher>
- https://de.wikipedia.org/wiki/Last_In_\T1\textendash_First_Out





12. Kugelspiel

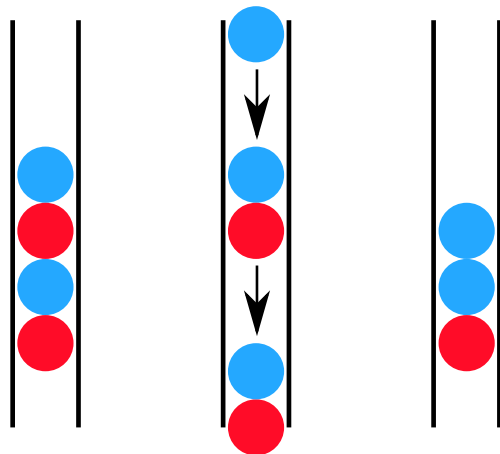
Emil spielt ein neues Spiel auf seinem Computer. Das Spiel beginnt mit einem Stapel aus mindestens drei farbigen Kugeln (rot oder blau), die sich in einer Röhre befinden.

Nach einem Klick auf eine Taste fallen die beiden jeweils unteren Kugeln aus der Röhre. Ausserdem fallen von oben neue Kugeln auf den Stapel. Abhängig von der Farbe der bisher untersten Kugel können zwei Dinge geschehen:

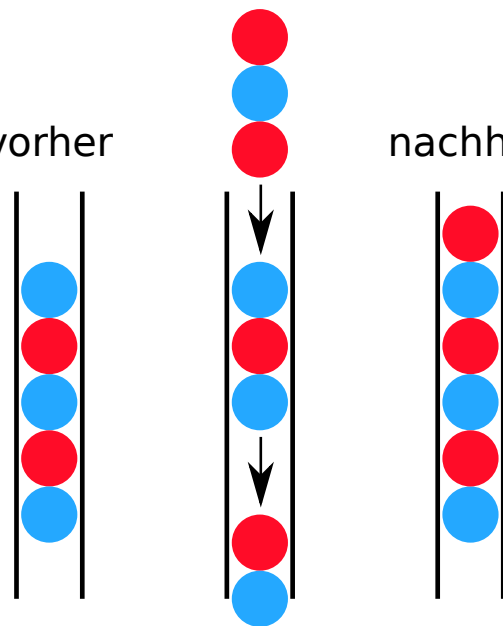
Falls die bisher unterste Kugel rot war, fällt eine blaue Kugel auf den Stapel:

Falls die bisher unterste Kugel blau war, fallen drei Kugeln mit den Farben rot, blau und rot auf den Stapel:

vorher nachher



vorher nachher



Solange mindestens drei Kugeln in der Röhre sind, drückt Emil immer wieder auf die Taste. Das Spiel endet, wenn sich weniger als drei Kugeln in der Röhre befinden.

Falls Emil mit einem Stapel wie in der Figur rechts beginnt, bleiben nach fünfmaligem Drücken nur zwei blaue Kugeln übrig, und das Spiel ist beendet.

Ziehe von rechts passend Farben auf die drei Stapelplätze, so dass Du einen Startstapel hast, bei dem das Spiel niemals enden wird.

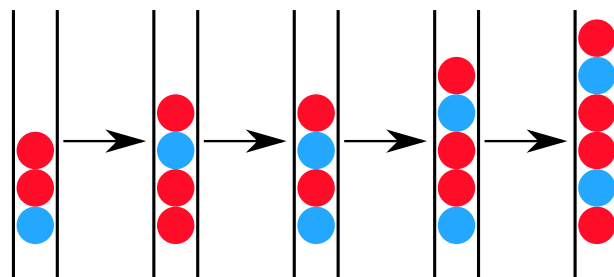
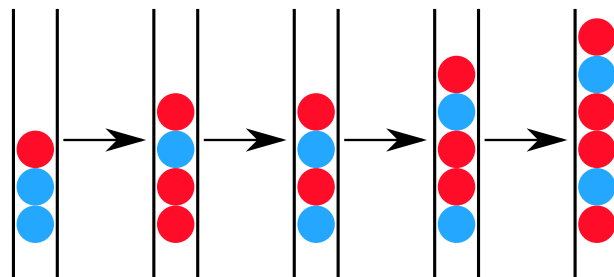
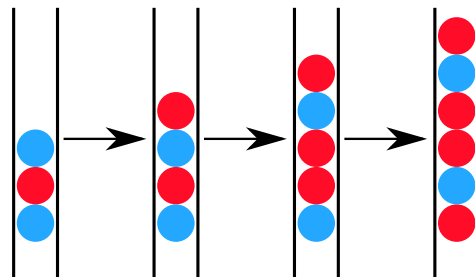
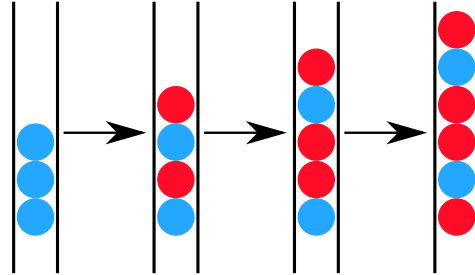




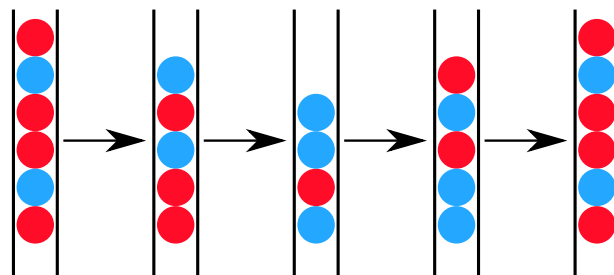
Lösung

Wenn die unterste Kugel eines Dreierstapels rot ist, endet das Spiel schon nach dem ersten Drücken: Dann sind nur noch zwei Kugeln in der Röhre.

Hingegen ist jeder Dreierstapel, dessen unterste Kugel blau ist, ein Endlos-Stapel. Mit höchstens viermaligem Drücken auf die Taste wird aus jedem der vier möglichen Dreierstapel mit einer blauen Kugel als unterste der Sechserstapel rot-blau-rot-rot-blau-rot:



Danach verläuft das Spiel immer in einem Viererzyklus:





Dies ist Informatik!

Das vorgestellte Spiel wurde als Beispiel von Emil Leon Post verwendet, um zu zeigen, dass es bei einer Verarbeitung von Zeichenketten (Strings) zu nicht abbrechenden Prozessen kommen kann. Emil Leon Post (1897-1954) war ein Polnischer Mathematiker und Logiker, welcher viele wissenschaftliche Beiträge zur theoretischen Informatik auf dem Gebiet der Aussagenlogik veröffentlichte.

Ein Ersetzungssystem wie in diesem Beispiel lässt sich mit Hilfe einer formalen Grammatik beschreiben. Dabei werden neben dem Eingabealphabet Regeln definiert, nach denen ersetzt wird. In unserem Fall wären beispielsweise die beiden vier Regeln spannend (wobei X für eine beliebige Kombination von blauen oder roten Kugeln steht):

$$Xbb \rightarrow rbrX$$

$$Xrb \rightarrow rbrX$$

$$Xbr \rightarrow bX$$

$$Xrr \rightarrow bX$$

Webseiten und Stichwörter

Rechenmodell, Formale Sprache, Produktionsregeln, Zeichenketten, verarbeitendes System

- http://esolangs.org/wiki/Post_canonical_system
- https://en.wikipedia.org/wiki/Tag_system
- https://en.wikipedia.org/wiki/Post_canonical_system



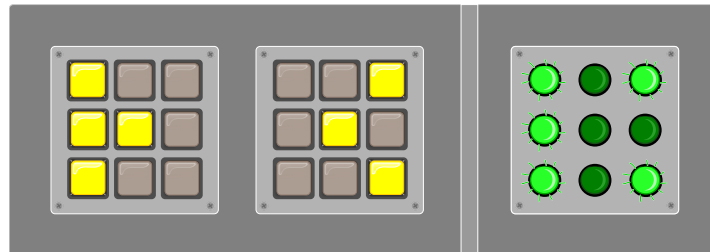


13. Zwei Möglichkeiten

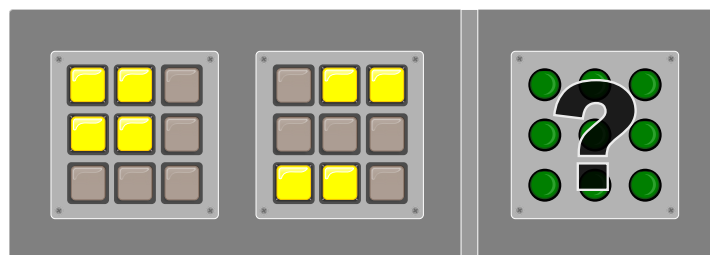
Erich hat ein altes elektronisches Geräte gefunden. Auf der linken Seite hat es zwei Felder mit je 9 Tasten, die man drücken kann. Auf der rechten Seite hat es ein Feld mit 9 Lampen. Je nachdem wie die Tasten gedrückt werden, gehen die Lampen an oder aus.

Erich beobachtet nun, dass die Position einer Lampe, die an- oder ausgeht, dieselbe ist, wie eine entsprechende Tastenkombination in den zwei Feldern.

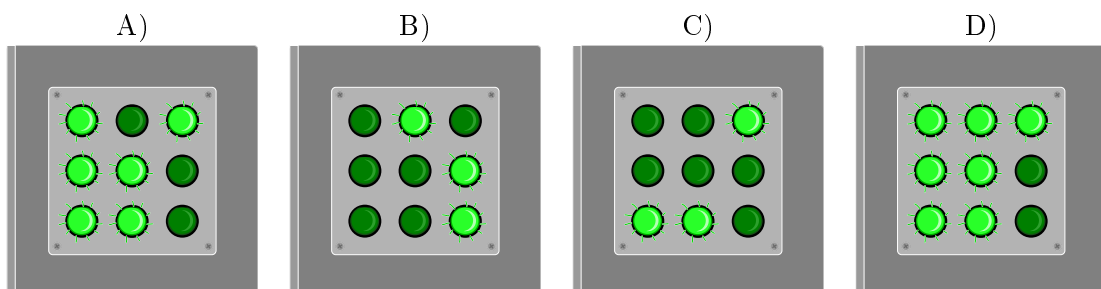
Momentan leuchten die Lampen wie folgt:



Erich ändert nun die Kombinationen, so dass sie so aussehen:



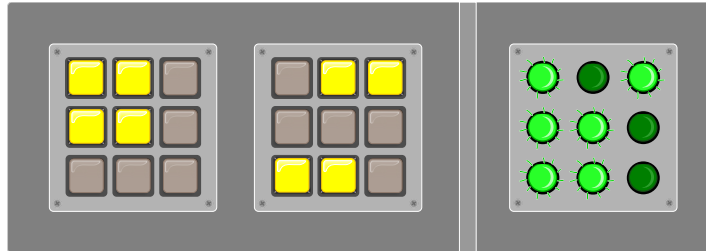
Wie leuchten die Lampen rechts?





Lösung

Die Regel ist: Wenn auf den beiden Eingabefeldern die gleiche Taste genau einmal eingeschaltet ist, dann leuchtet die entsprechende Lampe im Ausgabefeld, ansonsten leuchtet sie nicht.



Dies ist Informatik!

In allen Bereichen der Informatik wird gerne zweiwertige Logik benutzt, um über die Wahrheit und Falschheit von Aussagen innerhalb einer Anwendungssituation zu entscheiden. Das geschieht immer unter der Voraussetzung, dass es dafür genau zwei Möglichkeiten (zwei Werte) gibt. Eine dritte Möglichkeit ist ausgeschlossen („tertium non datur“). Eine logische Funktion liefert also entweder den Wahrheitswert „falsch“ oder den Wahrheitswert „wahr“.

In dieser Aufgabe benutzen wir die Funktion XOR („ausschliessendes Oder“, englisch: „exclusive or“), weil sie besonders häufig vorkommt. XOR funktioniert so: Wenn von zwei Aussagen genau eine „wahr“ ist und die andere „falsch“, dann liefert XOR den Wahrheitswert „wahr“. In den anderen Fällen liefert XOR den Wahrheitswert „falsch“. Im Beispiel zu dieser Aufgabe wird die logische Funktion XOR neunmal (Tasten) mit zwei Aussagen (Felder) vorgeführt.

Webseiten und Stichwörter

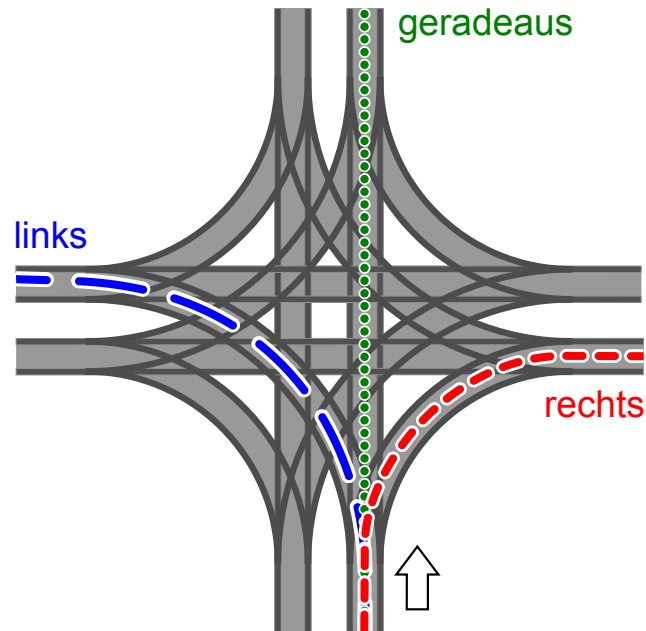
Logik, Zweiwertigkeit, Funktion XOR

- <https://de.wikipedia.org/wiki/Kontravalenz>
- <https://de.wikipedia.org/wiki/XOR-Gatter>



14. Tramkreuzung

Auf den Strassen von Sankt Petersburg fahren Trams. Dort gibt es eine Tramkreuzung, an der alle Wege möglich sind: Die Trams können aus jeder Richtung kommen und können jeweils gradeaus fahren, nach links oder nach rechts abbiegen.



Die Richtung, in die die Trams fahren, ist durch die Stellung der Weichen bestimmt. Die Stellungen der Weichen für Trams werden durch eine Kombination aus den Wörtern GERADEAUS, LINKS und RECHTS beschrieben. Beispiel: Die Kombination LINKS-GERADEAUS-LINKS-RECHTS bedeutet, dass die Weichen so gestellt sind, dass ein Tram nach links abbiegt, im Uhrzeigersinn das nächste Tram gradeaus fährt, das wieder im Uhrzeigersinn nächste Tram links fährt und das vierte Tram rechts fährt.

Nun kann es vorkommen, dass aus allen Richtungen Trams gleichzeitig an der Kreuzung ankommen. Welche Weichenstellungen können zu einem Zusammenstoss führen?

- A) RECHTS-RECHTS-RECHTS-RECHTS
- B) RECHTS-RECHTS-LINKS-LINKS
- C) LINKS-RECHTS-LINKS-RECHTS
- D) RECHTS-LINKS- RECHTS-LINKS



Lösung

Die richtige Antwort ist B). Das erste Tram (das nach rechts abbiegt) wird mit dem dritten Tram (das nach links abbiegt) zusammenstossen. Ebenso wird das zweite Tram (das nach rechts abbiegt) mit dem vierten Tram (das nach links abbiegt) zusammenstossen.

Bei allen anderen Lösungen fahren die Trambahnen kollisionsfrei aneinander vorbei: bei A) sind alle in der äusseren Kurve, bei C) und D) fahren immer zwei Trams nebeneinander vorbei durch dieselben Strassen. Bei C) und D) ist es noch wichtig zu sehen, dass die Kurven im Inneren sich nicht stören. Das nennt man „tangenciales“ oder auch „amerikanisches“ Linksabbiegen.

Dies ist Informatik!

Die Tramschienen – vor allem im Bereich von Kreuzungen – sind ein Beispiel für Ressourcen, die von mehreren Akteuren gemeinsam genutzt werden. Gleichzeitig darf aber an einer Stelle nur ein Tram sein ... sonst gibt es eine Kollision. Die Koordination und Optimierung der gemeinsamen Nutzung von Ressourcen durch parallel laufende Prozesse ist ein wichtiges Gebiet der Informatik. Wie bei der Trambahn muss man auch in der Digitaltechnik potenzielle Risiken beachten und Regeln für den sicheren Zugriff auf Datenbestände oder gemeinsam genutzte Geräte finden. Sonst droht der Verlust von Daten oder die Zerstörung von Hardware.

Im Fall der Tramkreuzung ist es eine gute Idee, nur sichere Weichenstellungen zuzulassen. Das kann man dort mit Hilfe von Signalen lösen. In der Informatik gibt es ebenfalls „Signal“-Lösungen für solche Probleme: sogenannte „Semaphoren“ (engl. für “Signal”) zeigen an, dass bestimmte Ressourcen belegt sind.

Webseiten und Stichwörter

Tramkreuzung, Prozesssynchronisation, Zugriff auf beschränkte Ressourcen

- <https://de.wikipedia.org/wiki/Prozesssynchronisation>
- [https://de.wikipedia.org/wiki/Abbiegen_\(Stra%C3%9Fenverkehr\)#Tangenciales_und_nichttangenciales_Abbiegen](https://de.wikipedia.org/wiki/Abbiegen_(Stra%C3%9Fenverkehr)#Tangenciales_und_nichttangenciales_Abbiegen)
- <http://www.swisseduc.ch/informatik/infotraffic/logictraffic/>



15. Codierung von Flaggen

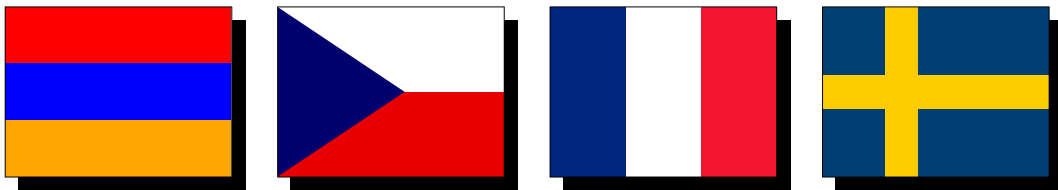
Computerbilder bestehen aus Zeilen mit Bildpunkten (Pixel). Wenn Computerbilder als Dateien gespeichert werden, wird im einfachsten Fall die Farbe jedes Pixels einzeln beschrieben. Mit dem (erfundenen) Dateiformat GIW werden Computerbilder komprimiert, also mit geringerer Dateigrösse gespeichert. Das funktioniert so:

- Jede Pixelzeile wird einzeln beschrieben.
- Jede Farbe wird durch ein Kürzel aus drei Buchstaben beschrieben.
- Eine Folge gleichfarbiger Pixel wird durch ein Klammerpaar beschrieben, das ein Farbkürzel und die Anzahl der gleichfarbigen Pixel enthält.

Eine Pixelzeile zum Beispiel, die durch die beiden Klammerpaare (grü,20)(wei,13) beschrieben wird, enthält zuerst 20 grüne und danach 13 weisse Pixel.

Unten siehst du vier Computerbilder von Flaggen. Die Bilder bestehen alle aus gleich vielen Pixelzeilen mit jeweils gleich vielen Pixeln. Sie wurden als Dateien im GIW-Format gespeichert.

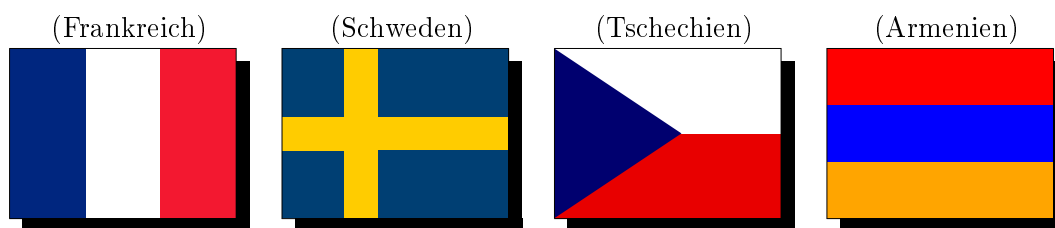
Ordne die Bilder nach der Grösse ihrer GIW-Datei!





Lösung

Die richtige Reihenfolge ist:



Falls eine ganze Zeile aus einer gleichen Farbe besteht, ist nur ein Klammerpaar zur Beschreibung nötig. Für jeden Farbwechsel in der Zeile kommt ein weiteres Klammerpaar hinzu. Beispielsweise hat die Flagge von Armenien in jeder Zeile die selbe Farbe, damit kann jede Zeile durch genau ein Klammerpaar kodiert werden.

Auf der anderen Seite wechselt bei der Flagge von Tschechien die Farbe innerhalb jeder Zeile genau einmal, von blau nach weiss oder von blau nach rot. Eine komprimierte Beschreibung dieser Flagge benötigt jeweils zwei Klammerpaare pro Zeile. Die französische Flagge wechselt in jeder Zeile zweimal die Farbe und benötigt folglich drei Klammerpaare pro Zeile für die komprimierte Beschreibung einer Zeile.

Die schwedische Flagge hat einen waagerechten gelben Streifen und benötigt für diesen Bereich ein Klammerpaar pro Zeile. In allen anderen Zeilen wechselt die Farbe zweimal und benötigt jeweils drei Klammerpaare. Daraus ergibt sich, dass die schwedische Fahne mehr Klammerpaare als die armenische, jedoch weniger als die französische braucht.

Wir müssen nun die schwedische Flagge mit der tschechischen Flagge vergleichen. Hätte die schwedische Flagge gleich viele Zeilen mit einer Farbe, wie mit drei Farben, dann bräuchte es im Mittel genau 2 Klammerpaare für die Beschreibung. Der mittlere Streifen der schwedischen Flagge ist jedoch schmaler als der restliche Teil der Flagge, so dass durchschnittlich mehr als zwei Klammerpaare für die Beschreibung benötigt werden. Daraus ergibt sich, dass die schwedische Flagge insgesamt mehr Klammerpaare als die tschechische Flagge benötigt.

Dies ist Informatik!

Die Datenkomprimierung ist ein wichtiger Teil der Informatik. Dank einer Reduktion von Speicherplatz lassen sich digitale Elemente oder Objekte mit weniger Zeitaufwand durch ein Netzwerk transportieren. Datenkomprimierungsalgorithmen können den Transferaufwand innerhalb eines Netzwerks signifikant verringern. Würde beispielsweise die Musik eines Webradios ohne Komprimierung versendet, so würde die zehnfache Datenmenge anfallen als bei heutzutage typischer Komprimierung. Aus diesem Grund wird intensiv an neuen Datenkomprimierungsalgorithmen geforscht, welche Fotos, Musik und Videos effizienter speichern können.

Das in dieser Aufgabe vorgestellte Komprimierungsverfahren gehört zum Typ der Lauflängenkodierung. Weitere Informationen kann das folgende YouTube Video liefern: https://www.youtube.com/watch?v=ydpNscvym_E.

Webseiten und Stichwörter

Kodierung, Komprimierung, Komprimierungsalgorithmus, Bitmap-Grafik

- https://www.youtube.com/watch?v=ydpNscvym_E



- <https://de.wikipedia.org/wiki/Lauflängenkodierung>



A. Aufgabenautoren

 Guðjón Karl Arnarson
 Wilfried Baumann
 Andrea Brabcová
 Eugenio Bravo
 Nicolas Brunner
 Christian Datzko
 Susanne Datzko
 Olivier Ens
 Jürgen Frühwirth
 Haris Gavranovic
 Martin Guggisberg
 Urs Hauser
 Hans-Werner Hein

 Mathias Hiron
 Juraj Hromkovič
 Akiko Kikui
 Tobias Kohn
 Ivana Kosírová
 Greg Lee
 Hiroki Manabe
 Tom Naughton
 Erkulan Nurtazanov
 Henry Ong
 Serena Pedrocchi
 Wolfgang Pohl
 Ilya Posov

 Sergei Pozdniakov
 Dániel Pressing
 Lorenzo Repetto
 Kirsten Schlüter
 Eljakim Schrijvers
 Sue Sentance
 Björn Steffen
 Seiichi Tani
 Jiří Vaníček
 Troy Vasiga
 Michael Weigend



B. Sponsoring: Wettbewerb 2016

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.roborobo.ch/>

Die RoboRobo Produkte fördern logisches Denken, Vorstellungsvermögen, Fähigkeiten Abläufe und Kombinationen auszudenken und diese systematisch aufzuzeichnen.

Diese Produkte gehören in innovative Schulen und fortschrittliche Familien. Kinder und Jugendliche können in einer Lektion geniale Roboter bauen und programmieren. Die Erwachsenen werden durch die Erfolgserlebnisse der „Erbauer“ miteinbezogen.

RoboRobo ist genial und ermöglicht ein gemeinsames Lern-Erlebnis!



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

digitec ist der Online-Marktführer der Schweiz. Egal, ob Fernseher, Smartphones oder Grafikkarten – bei digitec findest du alles rund um IT, Unterhaltungselektronik und Telekommunikation. Überzeuge dich selbst von der grossen Auswahl und stöbere in über 100'000 Produkten zu den besten Preisen.



<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/>



Kanton Zürich
Volkswirtschaftsdirektion
Amt für Wirtschaft und Arbeit

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



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

bbv Software Services AG ist ein Schweizer Software- und Beratungsunternehmen. Wir stehen für Top-Qualität im Software Engineering und für viel Erfahrung in der Umsetzung. Wir haben uns zum Ziel gesetzt, unsere Expertise in die bedeutendsten Visionen, Projekte und Herausforderungen unserer Kunden einzubringen. Wir sind dabei als Experte oder ganzes Entwicklungsteam im Einsatz und entwickeln individuelle Softwarelösungen.

Im Bereich der Informatik-Nachwuchsförderung engagiert sich die bbv Software Services AG sowohl über Sponsoring als auch über die Ausbildung von Lehrlingen. Wir bieten Schnupperlehrtage an und bilden Informatiklehrlinge in der Richtung Applikationsentwicklung aus. Mehr dazu erfahren Sie auf unserer Website in der Rubrik Nachwuchsförderung.



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

Beratung ist keine Nebensache

Wir interessieren uns, warum, wann und wie die Werbeartikel eingesetzt werden sollen – vor allem aber, wer angesprochen werden soll.



<https://www.hslu.ch/de-ch/informatik/agenda/veranstaltungen/fuer-schulen/itgirls/>

HLSU, Lucerne University of Applied Sciences and Arts
Engineering & Architecture



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

Pädagogische Hochschule Luzern



ABZ

AUSBILDUNGS- UND BERATUNGSZENTRUM
FÜR INFORMATIKUNTERRICHT

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

Ausbildungs- und Beratungszentrum für Informatikunter-
richt der ETH 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.



Der Informatik-Biber neu auf Facebook:

<https://www.facebook.com/informatikbiberch>

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001



www.svia-ssie-ssii.ch
schweizerischervereinfürinformatikind
erausbildung//sociétésuissedel'inform
atique dans l'enseignement//societàsviz
zeraperl'informaticanell'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.