



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

Aufgaben und Lösungen 2017 Alle Stufen

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

Herausgeber:
Christian Datzko, Hanspeter Erni

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001

SV!A

www.svia-ssie-ssii.ch
schweizerischervereinfürinformatikin
erausbildung//sociétésuisse
del'informatique
dansl'enseignement//società
svizzeraperl'informaticanell'
insegnamento



Mitarbeit Informatik-Biber 2017

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

Herzlichen Dank an:

Juraj Hromkovič, Giovanni Serafini, Urs Hauser, Regula Lacher, Ivana Kosírová: ETHZ

Valentina Dagienė: Bebras.org

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

Anna Morpurgo, Violetta Lonati, Mattia Monga: Italien

Gerald Futschek, Wilfried Baumann: Oesterreichische Computer Gesellschaft, Österreich

Zsuzsa Pluhár: ELTE Informatikai Kar, Ungarn

Eljakim Schrijvers, Daphne Blokhuis: 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 2017 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 2017 geprüft. Dieses Aufgabenheft wurde am 18. November 2017 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. 101 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 2017 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 2017	i
Vorwort	ii
1. Parkplatz	1
2. Vogelhäuser	3
3. Finde die Lücke	5
4. Binäres Gartentor	7
5. Nachrichtendienst	9
6. Erdbeerjagd	11
7. Tauschen von Hunden	13
8. Der einarmige Biber	15
9. Mauern entfernen	19
10. Fünf Hölzli	23
11. Dancing like the Stars	25
12. Japanischer Ehrenname	29
13. Kurzes Programm	31
14. Klammeranhänger	33
15. Schülerzeitung	35
16. Honomakato	39
17. Japanischer Stockkampf	43
18. Opas Marmelade	45
19. Stadt der Kreisel	47
20. Pizzeria Biberia	49
21. Geheime Bestellung	51
22. Münzdreh	53
23. „Saftladen“	55
24. Alarm im Museum!	59

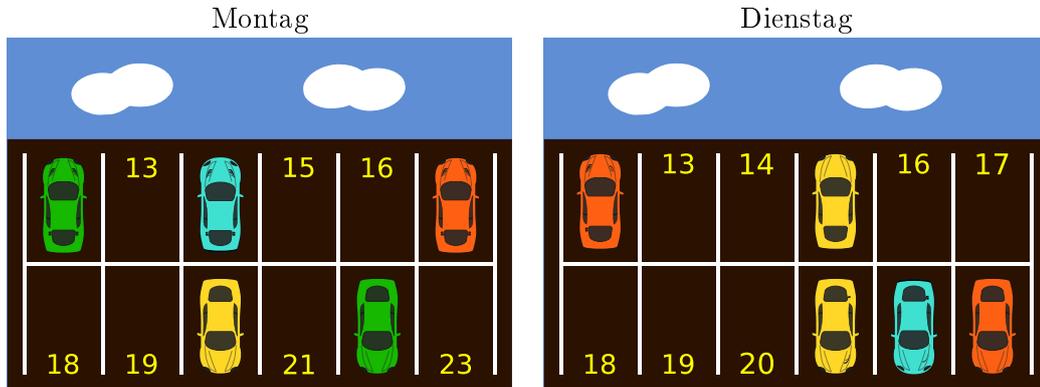


25. Lichtkunst	61
26. Ersetzungen	63
27. Wege durch den Irrgarten	65
28. Bewässerungssystem	67
29. Der neue Song	69
30. Kugelbahn	71
31. Gänge im Biberbau	75
32. Hilf dem Arabot!	79
33. Zahnstocherteilete	81
34. Wortabstände	83
35. Mehrere Downloads	85
36. Punkte sammeln	89
37. Bildviertel	93
38. Abkürzung oder Umweg?	95
39. Digitalziffern	97
40. Zerteile die Ziffernfolge	99
A. Aufgabenautoren	101
B. Sponsoring: Wettbewerb 2017	102
C. Weiterführende Angebote	105



1. Parkplatz

Auf dem Biberplatz hat es 12 Parkplätze für Autos. Jeder Parkplatz ist mit einer Nummer gekennzeichnet. Die Bilder unten zeigen, welche Parkplätze am Montag und welche am Dienstag besetzt waren.



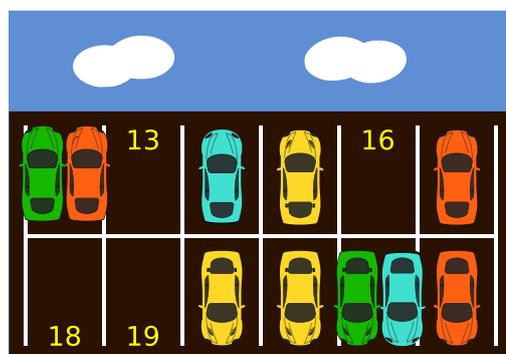
Wie viele Parkplätze waren sowohl am Montag als auch am Dienstag frei?

- A) 3
- B) 4
- C) 5
- D) 6



Lösung

Die Parkplätze 13, 16, 18 und 19 waren sowohl am Montag als auch am Dienstag frei. Wenn die Autos von beiden Tagen auf einem Parkplatz dargestellt werden, sehen wir, welche Parkplätze frei bleiben. Diese Parklücken können dann einfach gezählt werden. Am Montag und Dienstag waren insgesamt 4 Parkplätze durchgehend frei.



Dies ist Informatik!

Alle Arten von Daten können als eine Folge von Nullen und Einsen codiert werden. Jede Null oder Eins wird ein Bit genannt. Eine solche Sequenz heisst Binärcode.

Bei dieser Aufgabe können wir die Anwesenheit eines Autos auf einem Parkplatz mit einer Eins (1) und eine Abwesenheit als einer Null (0) codieren. So lässt sich die Belegung des Parkplatzes als Reihenfolge von Bits modellieren. Der Montag entspricht 101001001010, der Dienstag entspricht 100100000111. Mit Hilfe einer logischen ODER (OR) Verknüpfung werden alle Parkplätze, welche mindestens an einem der beiden Tage belegt waren mit einer 1 gekennzeichnet. Die richtige Antwort berechnen wir, indem wir die Reihen übereinander legen. Wir sehen, dass aus 101001001010 ODER (OR) 100100000111 schlussendlich 101101001111 wird.

$$\begin{array}{r}
101001001010 \\
\text{oder} \\
100100000111 \\
= \\
101101001111
\end{array}$$

Diese Binärzahl enthält 4 Nullen, also 4 freien Parkplätzen.

Webseiten und Stichwörter

Bits, Binär, ODER, logischer Operator

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

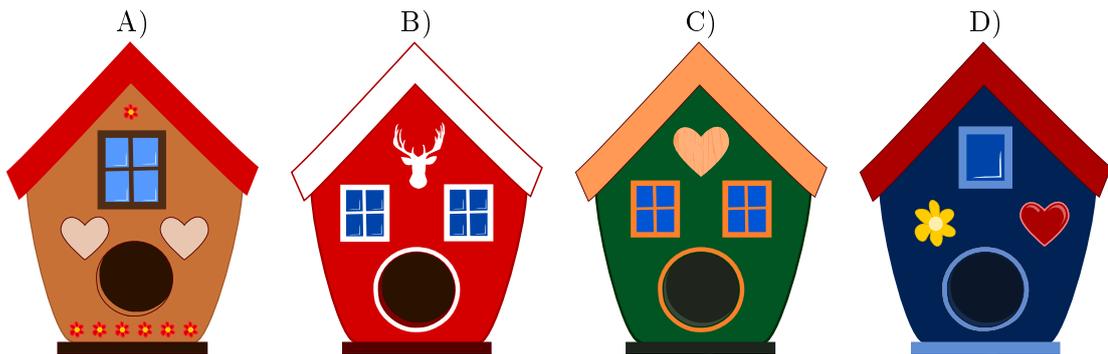


2. Vogelhäuser

Die Biber Mutter will ein Vogelhaus als Geschenk für den Geburtstag ihrer Tochter kaufen. Sie fragte die Tochter, was für ein Vogelhaus sie gerne hätte. Ihr Tochter antwortete:

„Ich möchte ein Vogelhaus mit 2 Fenstern und einem Herzen.“

Also geht die Biber Mutter zum Tiershop um das passende Vogelhaus zu kaufen.
Welches ist das Haus, das die Biber Mutter für ihre Tochter kaufen kann?





Lösung

Die Antwort C) ist richtig, denn das Haus 2 Fenster und 1 Herz hat.

Die Antwort A) ist falsch, weil Haus nur ein Fenster und 2 Herzen hat.

Die Antwort B) ist falsch, weil Haus zwei Fenster, aber kein Herz hat.

Die Antwort D) ist falsch, das Haus nur ein Fenster hat.

Dies ist Informatik!

Die Schwierigkeit dieser Aufgabe ist, bei den vier Häusern nur die beiden Wünsche der Tochter zu berücksichtigen. Die Farbe des Hauses oder weitere Besonderheiten sind egal. Man muss also unwichtige Dinge ignorieren.

Dieser eingeschränkte Blick ist eine Abstraktion, man schaut nur auf bestimmte Eigenschaften. In der Praxis muss man häufig den Blick auf das Wesentliche richten.

Webseiten und Stichwörter

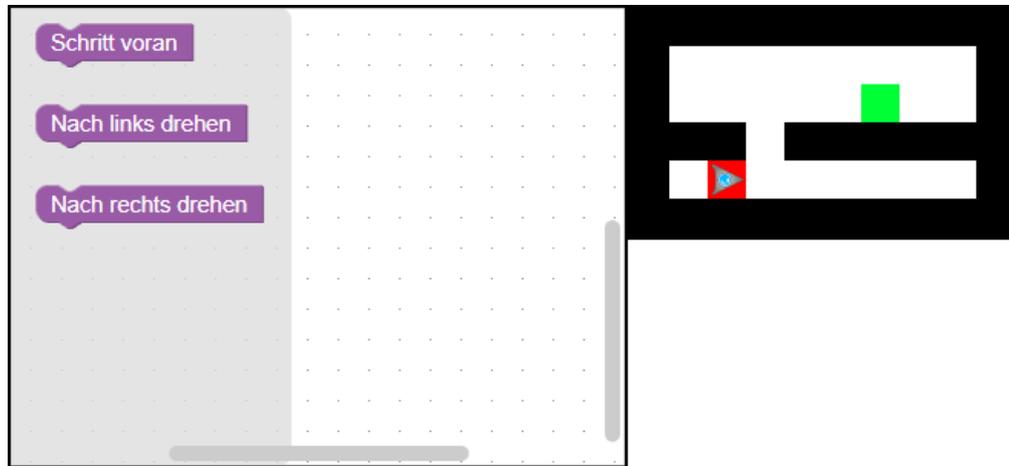
Muster, Mustererkennung, Eigentum, Abstraktion

- <https://de.wikipedia.org/wiki/Abstraktion>
- <https://de.wikipedia.org/wiki/Mustererkennung>



3. Finde die Lücke

Ein dreieckiger Roboter steht auf dem roten Startfeld und soll zum grünen Zielfeld fahren.

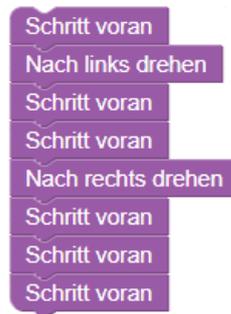


Programmiere den Roboter, indem du die Befehle in der richtigen Reihenfolge für das Programmfenster wählst.



Lösung

Eine mögliche Antwort ist:



Damit der Roboter von seinem Startfeld auf das Zielfeld gelangt, muss er zunächst ein Feld nach vorne fahren, damit er neben der Lücke in der Mauer steht. Wenn er sich nach links dreht und zwei Felder nach vorne fährt, ist er durch die Lücke in der Mauer durch. Nun braucht er sich nur noch nach rechts zu drehen und drei Felder nach vorne zu fahren, um beim Zielfeld anzukommen.

Dies ist Informatik!

In der mobilen Robotik ist das Navigieren ein allgemeines Problem. Labyrinth-Lösungen kommen zwar nicht häufig vor, aber erfordern ähnliche Computational-Thinking-Fähigkeiten. Um dieses Problem zu lösen wird ein automatisierter Roboter verwendet. Labyrinth können von unterschiedlicher Art sein: Mit oder ohne Schleifen, mit oder ohne Rastersystemen. Der Roboter dieser Aufgabe hat keine Sensoren, um beispielsweise Mauern zu erkennen. Er ist quasi „blind“. Daher muss er mit ganz präzisen Anweisungen versorgt werden, damit er die Lücke und somit den korrekten Weg findet.

Webseiten und Stichwörter

Sequenz, Programmieren

- <https://en.wikipedia.org/wiki/Blockly>
- <http://primalogo.ch/>
- <http://www.abz.inf.ethz.ch/primarschulen-stufe-sek-1/programmieren-fur-kinder/>
- http://www.swisseduc.ch/informatik/programmiersprachen/scratch_werkstatt/
- <http://ilearnit.ch/de/2b/explain.html>
- <https://scratch.mit.edu/>
- <http://kinderlabor.ch/informatik-fuer-kinder/programmieren-mit-scratch/>



4. Binäres Gartentor

Die Biber besuchen einander oft. Aber ... manchmal ist man jedoch nicht zu Hause. Dann hinterlässt man an seinem Steinplatten-Gartentor eine Nachricht, wann man ungefähr wieder da ist. Dazu steckt man bis zu drei Holzstäbe gerade zwischen gegenüberliegende Bohrlöcher der Steinplatten.

Die Biber haben diese vier Gartentor-Nachrichten miteinander verabredet:



Wir sind da, kommt doch rein.



Wir werden schon mittags zurück sein.



Wir sind leider erst am Abend wieder da.



Wir besuchen jemanden und kommen um Mitternacht heim.

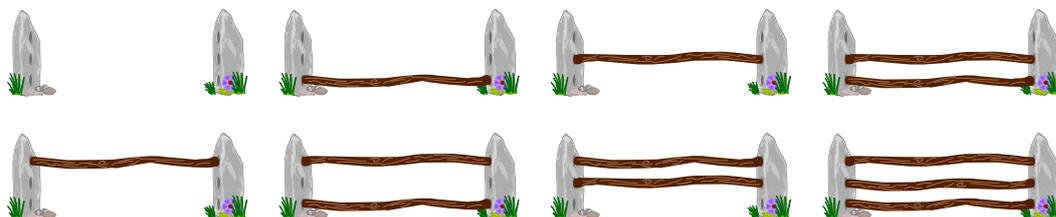
Die Biber könnten noch weitere Nachrichten verabreden, ohne zusätzliche Holzstäbe oder Bohrlöcher zu verwenden.

Wie viele verschiedene Nachrichten können sie insgesamt, also zusammen mit den vier Nachrichten oben, vereinbaren?



Lösung

Sie können insgesamt acht verschiedene Nachrichten vereinbaren:



Dies ist Informatik!

In dieser Biberaufgabe benutzen die Biber ein dreistelliges binäres Zeichensystem. Die Träger der Information sind die gegenüberliegenden Bohrloch-Paare. Ein solches Bohrloch-Paar hat zwei Zustände, und zwar „Holzstab gesteckt“ oder „Holzstab nicht gesteckt“. Die Position der Bohrloch-Paare gibt den Stellenwert wieder. Die Anzahl unterschiedlicher Nachrichten ist gleich der Anzahl der Bohrloch-Paar-Zustände (2) hoch der Anzahl der Bohrloch-Paare (3), Also: $2^3 = 2 \cdot 2 \cdot 2 = 8$. Die Biber wissen aus Tradition, was ihre freundlichen Nachrichten bedeuten, und wenn sie sich mal irren, kann kein grosser Schaden entstehen. In der Informatik mit ihren global vernetzten Systemen sind dagegen alle mit allen direkte Kommunikationsnachbarn und müssen sich fehlerfrei verstehen. Grosse Organisationen kümmern sich um die Standardisierung von Zeichensystemen und die Zertifizierung von Produkten. In multinational besetzten Fachausschüssen wird über das Aussehen von Zeichen und deren Bedeutung verhandelt und beschlossen. Die wichtigsten Zeichensysteme werden von Parlamenten vieler Staaten zu Gesetzen erklärt. So sorgen sich viele Menschen in aller Welt darum, dass sich alle Computer der Welt gegenseitig gut verstehen.

Webseiten und Stichwörter

Kodierung, Binärcode, Standard

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



5. Nachrichtendienst

Violet will mit Hilfe der Biber eine Nachricht an Leo schicken. Sie unterteilt die Nachricht in Abschnitte von 3 Buchstaben auf jeder Karte und gibt jedem Biber eine Karte mit diesen Buchstaben. Violet weiss, dass die Biber während dem Transport manchmal abgelenkt werden und daher zu verschiedenen Zeit ankommen. Daher nummeriert Violet die Karten, bevor sie diese den Bibern abgibt. Leo muss dann die Karten in der richtigen Reihenfolge anordnen um die vollständige Nachricht lesen zu können.

Um die Nachricht PARTYZEIT zu senden, erstellt Violet die drei Karten wie folgt:



Leo erhält nun die folgende Kartensequenz:



Wie lautet die Originalnachricht?

- A) NFUALLEDEHOLSSB
- B) ALLEDENFUSSBNFU
- C) HOLEDENFUSSBALL
- D) SSBALLHOLEDENFU



Lösung

Die richtige Antwort lautet: C) HOLEDENFUSSBALL.

Wenn man die Karten in der richtigen Reihenfolge ordnet, so erhält man folgende Originalnachricht:



Dies ist Informatik!

Wenn Daten wie zum Beispiel E-Mails, Bilder oder Videos im Internet verschickt werden, werden sie in kleine Pakete (TCP/IP-Paket) zerlegt. Jedes Paket hat eine Grösse von maximal von 65536 Zeichen ($2^{16} = 65536$ Zeichen = 64 KB).

Diese Pakete werden dann teilweise über verschiedene Router mit einigen zusätzlichen Informationen über das Paket selbst (Sender, Empfänger, Sequenznummer des Paketes, ...) gesendet. Alle diese Informationen sorgen dafür, dass die Originalinformation, auch wenn die Teilinformationen (Teilpakete) zeitversetzt ankommen, beim Empfänger wieder korrekt zusammengesetzt werden kann.

Anmerkung: Bei IPv6 sind grössere Pakete möglich.

Webseiten und Stichwörter

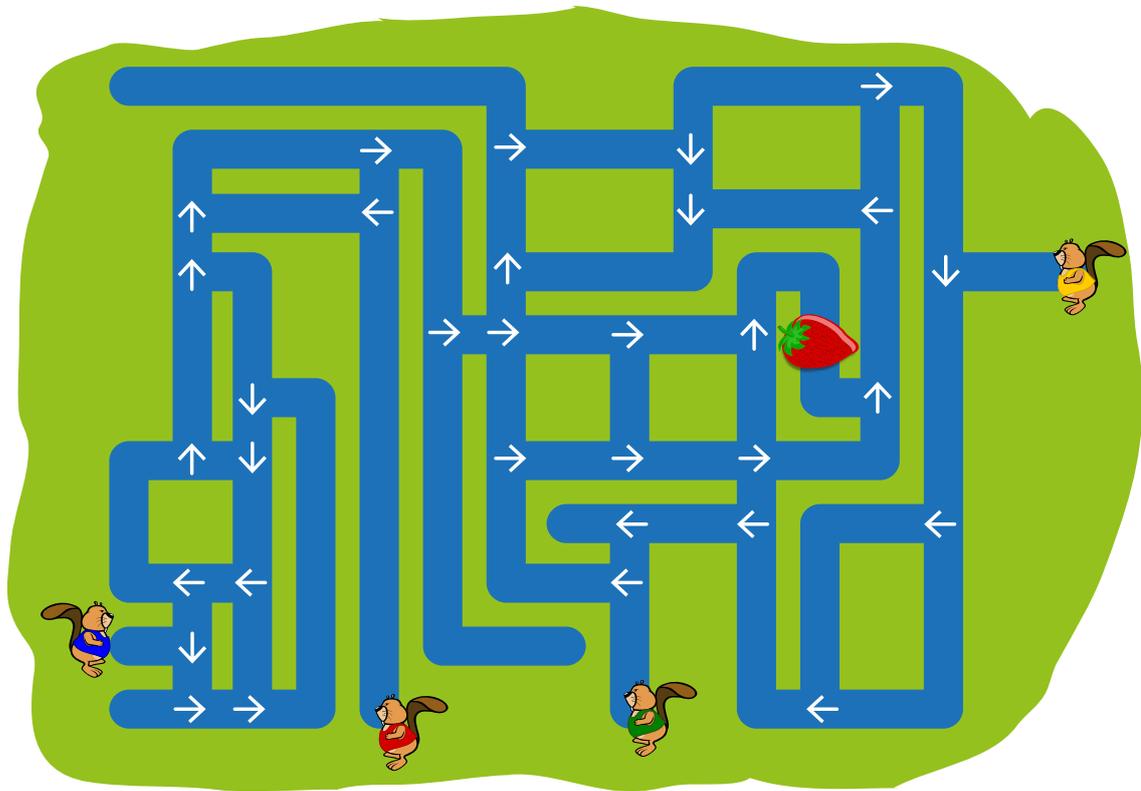
Internet-Protokoll, Pakete

- <https://de.wikipedia.org/wiki/IP-Paket>
- <https://de.wikipedia.org/wiki/IP-Fragmentierung>



6. Erdbeerjagd

Vier Biber fangen von vier verschiedenen Startpositionen an zu schwimmen. Sie schwimmen vorwärts und folgen den Pfeilen.



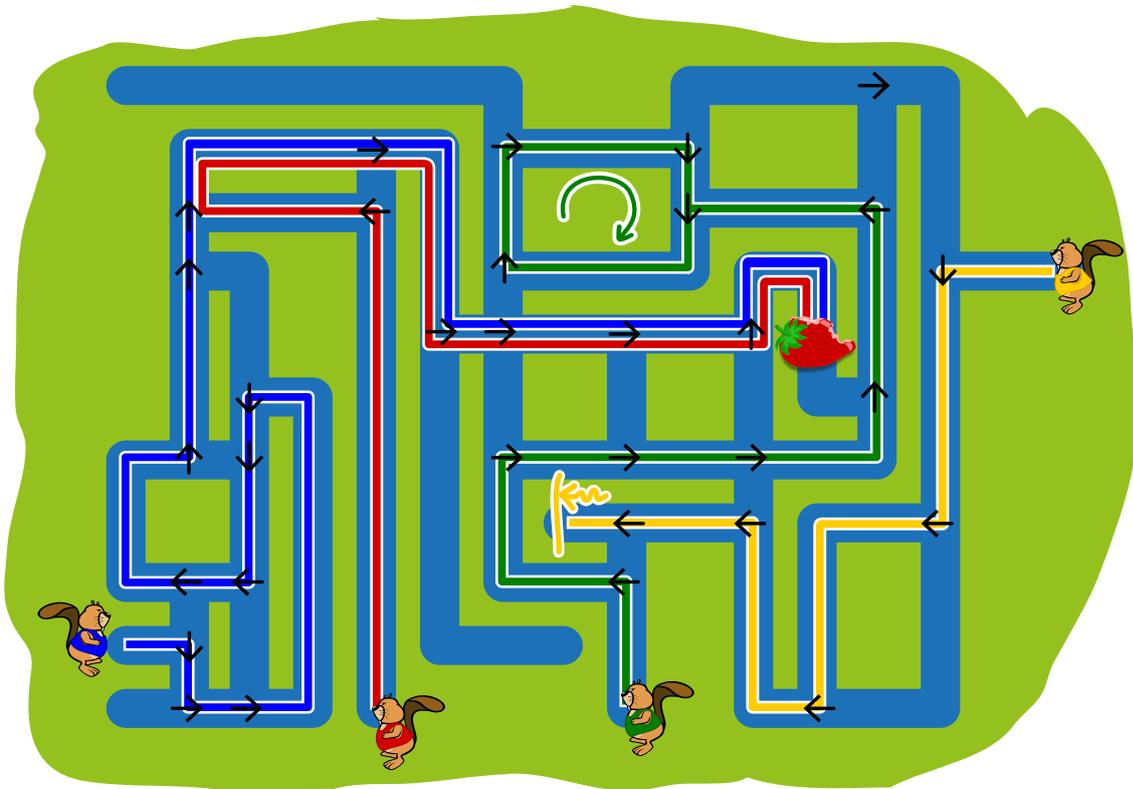
Wie viele Biber werden an der Erdbeere vorbei schwimmen?

- A) 0
- B) 1
- C) 2
- D) 3
- E) 4



Lösung

Zwei der vier Biber werden an der Erdbeere vorbei schwimmen, und zwar die beiden links unten:



Der dritte Biber von links wird letztlich immer im Kreis schwimmen und der Biber rechts oben wird in einer Sackgasse landen, aus der er nicht wieder heraus kommt.

Dies ist Informatik!

Bei dem Kanalsystem kann man zwischen zwei verschiedenen Elementen unterscheiden: Kanäle (durch die ein Biber einfach durchschwimmt) und Kreuzungen (an denen sich ein Biber anhand des Pfeiles entscheiden muss, wohin er weiter schwimmt). In der Informatik nennt man ein solches System einen *Graphen*, wobei die Kanäle *Kanten* genannt werden und die Kreuzungen *Knoten*. Kanten und Knoten können Informationen enthalten. In diesem Fall enthalten die Knoten die Information, welche Kante nach dem Knoten folgt.

Graphen können nicht nur für solche Aufgaben verwendet werden, auch zum Programmieren von Computern sind sie hilfreich. Ein Computer kann beispielsweise Anweisungen in einem Graphen folgen, damit er weiss, was er als nächstes machen soll. Manchmal schafft er es dabei, ein Problem zu lösen (bei der Erdbeere anzukommen), manchmal endet er in einer Sackgasse und manchmal leider auch in einer Endlosschleife, aus der er von alleine nicht wieder heraus kommt.

Webseiten und Stichwörter

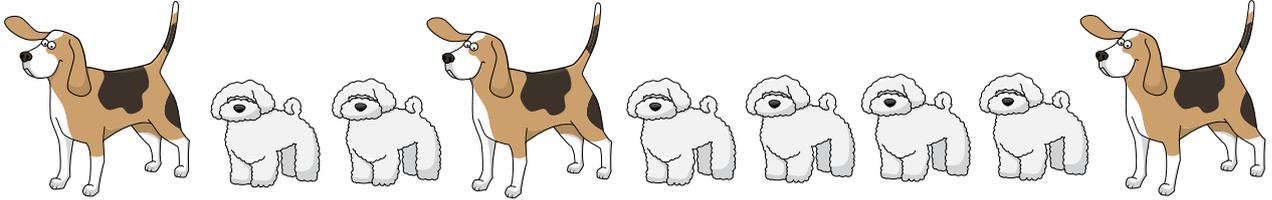
Graphen lesen

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

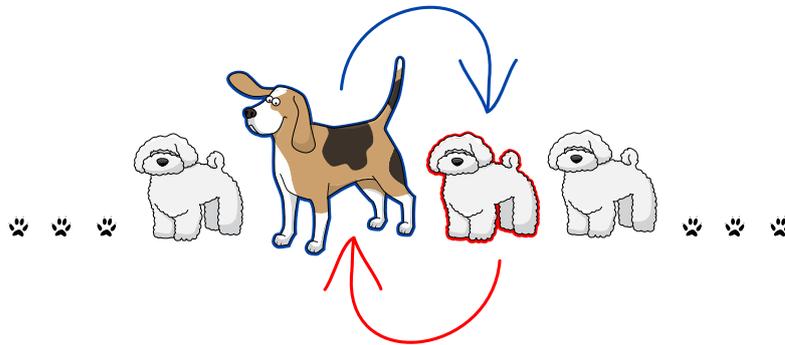


7. Tauschen von Hunden

Hunde aus zwei verschiedene Arten stehen in einer Reihe:



Ein Tausch findet statt, wenn 2 Hunde, die nebeneinander stehen, ihre Plätze tauschen:



Nach einigen Vertauschungen sind die drei grossen Hunde nebeneinander.

Was ist die kleinste Anzahl Vertauschungen, damit die drei grossen Hunde nebeneinander stehen?

- A) 5
- B) 6
- C) 7
- D) 8



Lösung

Die richtige Antwort lautet B) (6): Zuerst muss man den ersten grossen Hund zweimal nach rechts tauschen und dann den letzten grossen Hund viermal nach links.

Jeder kleine Hund wird dabei einmal getauscht, da jeder kleine Hund zwischen zwei grossen Hunden positioniert ist. Die Vertauschung von zwei kleinen Hunden hat keine Wirkung, so dass jeder nützliche Tausch zwischen einem kleinen Hund und einem grossen Hund stattfinden muss. Da es insgesamt 6 kleine Hunde gibt, bedeutet dies, dass es mindestens 6 Vertauschungen geben muss.

Will man alle grossen Hunden ganz nach rechts oder nach links positionieren, so erfordert das mehr als 6 Vertauschungen.

Dies ist Informatik!

Normalerweise könnten die Hunde ja einfach wild durcheinanderlaufen bis die drei grossen Hunde nebeneinander stehen. In dieser Aufgabe stehen die Hunde jedoch für Daten in einem Speicher.

Wenn ein Computer Daten sortieren soll, und er nicht viel Platz zum Zwischenspeichern hat, tauscht er in der Regel nur zwei Daten (Hunde) miteinander aus. Dies nennt man einen *swap*.

In diesem Fall soll der Computer möglichst schnell, also mit möglichst wenigen swaps die drei grossen Hunde zusammenbringen. Dies ist ebenfalls typisch für Computer: auch wenn einige wenige swaps schneller stattfinden als wir Menschen es mitbekommen können, müssen Computer mit sehr viel mehr Daten arbeiten. Das bedeutet, dass ein schnellerer Algorithmus Daten vielleicht innerhalb von einer halben Sekunde sortiert hat, anstelle dass er dazu zwei Minuten bräuchte. Bei ganz grossen Berechnungen kann der Unterschied tatsächlich zwischen „einige Tage rechnen“ und „mehrere Jahre rechnen“ liegen, wenn der Algorithmus gut oder schlecht programmiert ist.

Webseiten und Stichwörter

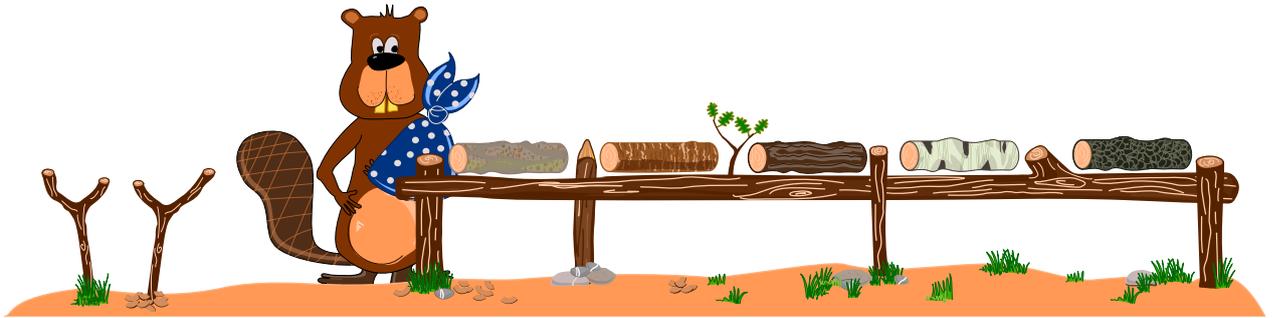
Variablentausch

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



8. Der einarmige Biber

Der arme David hat sich den Arm gebrochen und kann momentan nur den anderen verwenden. Er möchte seine Sammlung verschiedener Holzscheite sortieren, kann aber immer nur ein Holzschleit aufs Mal hochheben. Er kann ein Holzschleit auf dem zusätzlichen Halter von Dir aus gesehen links von ihm ablegen.

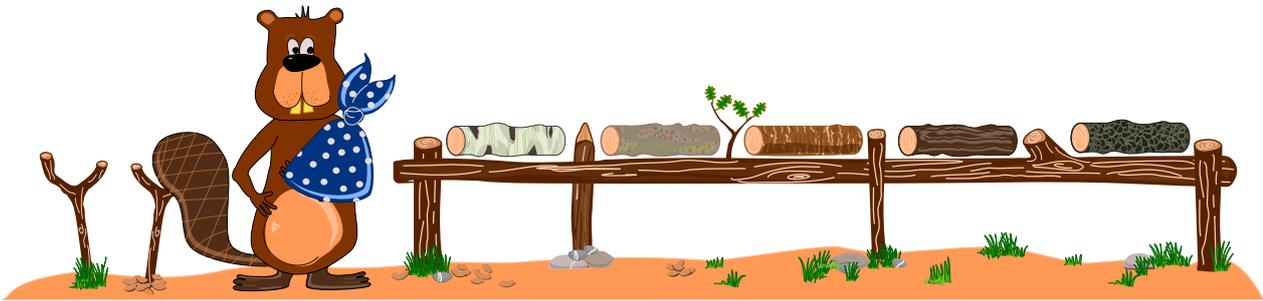


Hilf David, seine Sammlung verschiedener Holzschleit der Helligkeit nach zu sortieren, so dass das hellste Holzschleit von Dir aus gesehen links und das dunkelste Holzschleit von Dir aus gesehen rechts liegt.



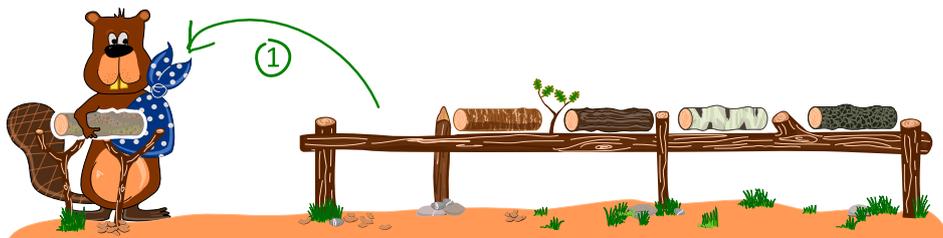
Lösung

Die richtige Antwort ist:

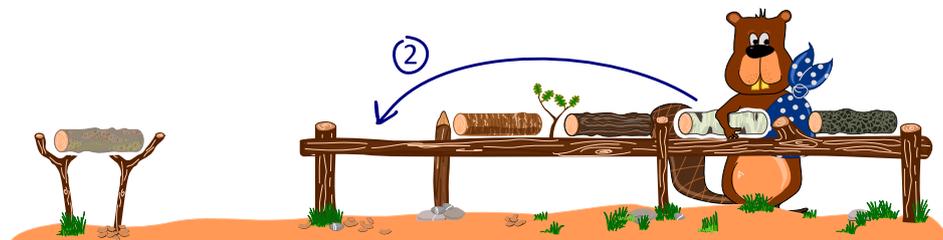


Damit David diese Aufgabe lösen kann, muss er den zusätzlichen Halter verwenden. Es gibt viele verschiedene Möglichkeiten, diesen Task korrekt zu lösen, aber am einfachsten ist es wohl, immer zwei Holzscheite mit Hilfe des zusätzlichen Halters auszutauschen:

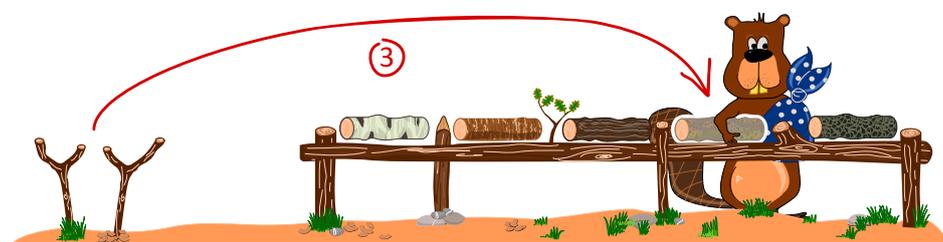
- Ein Holzschicht von seinem ursprünglichen Platz auf den zusätzlichen Halter legen.



- Ein anderes Holzschicht auf den freigewordenen Platz legen.



- Das Holzschicht vom zusätzlichen Halter auf den nun freigewordenen neuen Platz legen.



So kann man durch mehrfaches Vertauschen von jeweils zwei Holzschichten am Ende die richtige Ordnung hergestellt haben.



Dies ist Informatik!

Die Methode, zwei Holzscheite auf zwei Plätzen durch einen zusätzlichen dritten Platz auszutauschen, wird in der Informatik sehr häufig verwendet. Die „Plätze“ sind in der Regel Variablen und es geht darum, die Werte von zwei Variablen (der Einfachheit halber a und b genannt) auszutauschen. Dazu wird der Wert der Variablen a in eine temporäre Variable t geschrieben, der Wert der Variablen b in die Variable a und der Wert der temporären Variablen t in die Variable b . Man schreibt auch:

$$t \leftarrow a$$
$$a \leftarrow b$$
$$b \leftarrow t$$

Um ein Feld mit mehreren Variablen und nur einer temporären Variablen zu sortieren, kann man beispielsweise das *Sortieren durch Auswählen* verwenden:

- Von der ersten Variablen bis zur letzten Variablen:
 - Wähle jeweils den kleinsten Wert des noch nicht sortierten Feldes
 - Tausche diesen Wert mit dem Wert der aktuellen Variablen mit Hilfe der temporären Variablen aus

Sortieren durch Auswählen ist nicht das einzige einfache Sortierverfahren, häufig wird auch *Sortieren durch Einfügen* oder *Sortieren durch Aufsteigen* verwendet. Ein schnelleres aber auch komplizierteres Verfahren ist das *Quicksort-Verfahren*.

Webseiten und Stichwörter

Dreieckstausch (Tausch von Variablen), Sortieren

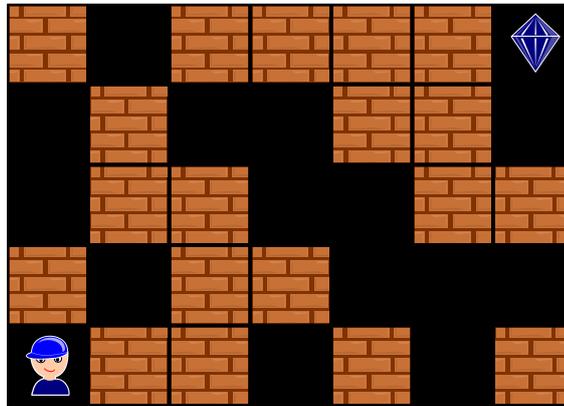
- <https://de.wikipedia.org/wiki/Dreieckstausch>
- <https://de.wikipedia.org/wiki/Sortierverfahren>





9. Mauern entfernen

Um zum Schatz in der rechten oberen Ecke zu kommen, muss Peter Mauern entfernen. Er will dabei in möglichst wenig Feldern Mauern entfernen.

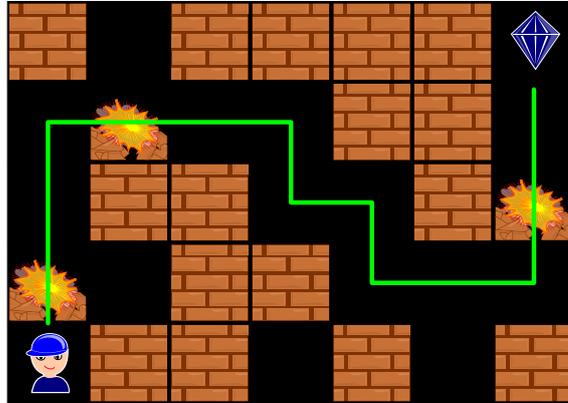


Entferne möglichst wenig Felder mit Mauern, sodass ein Weg zum Schatz frei wird.

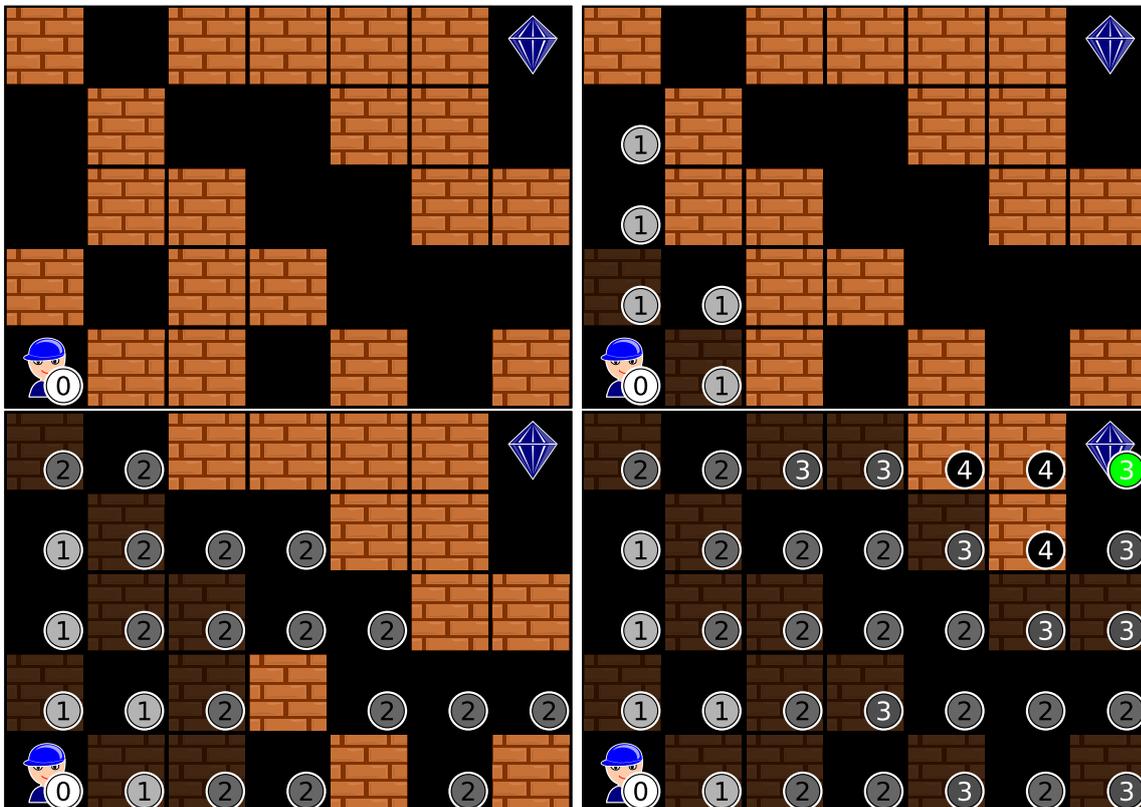


Lösung

Es müssen mindestens drei Mauern entfernt werden. Im Bild unten sind diese Mauern durch die Explosionen gekennzeichnet, der entstandene freie Weg ist grün dargestellt.



Systematisch kann man das Ergebnis erhalten, wenn man jede Zelle markiert mit der „Anzahl an Mauern, die notwendigerweise entfernt werden müssen, um dorthin zu kommen“. Wir beginnen in der Start-Ecke links unten. Wir markieren dieses Start-Feld mit 0. Alle weiteren schwarzen Felder, die man von dort direkt erreichen kann, markieren wir auch mit 0. Dann markieren wir die auf 0 angrenzenden Mauern mit 1, sowie alle schwarzen Felder, die man durch Entfernen einer dieser Mauern nun neu erreichen kann auch mit 1. Dann werden ebenso möglichst viele Mauern und schwarzen Felder mit 2 markiert, das sind dann alle Felder, die man durch Entfernen von 2 Mauern erreichen kann. Anschliessend markiert man alle Felder, die man durch Entfernen von 3 Mauern erreicht mit 3, usw. Wenn schliesslich das Zielfeld markiert ist, kennt man die kleinste Anzahl an zu entfernenden Mauern. In unserem Fall steht dort schliesslich die 3:





Dies ist Informatik!

Die Suche nach einem kürzesten Weg in einem Labyrinth ist ein bekanntes informatisches Problem. Auch Navigationssysteme nutzen verschiedene Techniken zum Auffinden kürzester Wege, auch unter bestimmten Nebenbedingungen wie dem Vermeiden von Strassen mit Gebühren (z.B. Autobahnen). In unserem Fall kommt es nur darauf an einen kürzesten Weg durch die Mauern zu nehmen, die Länge der anderen „normalen“ Wege sind in unserer Aufgabe zu vernachlässigen. Das Beantworten dieser Aufgabe erfordert algorithmisches Denken, es werden Zwischenergebnisse gespeichert und diese systematisch vom Startfeld ausgehend eingetragen. Man nennt diese Vorgangsweise auch Breitensuche.

Webseiten und Stichwörter

Labyrinth, Weg, Breitensuche

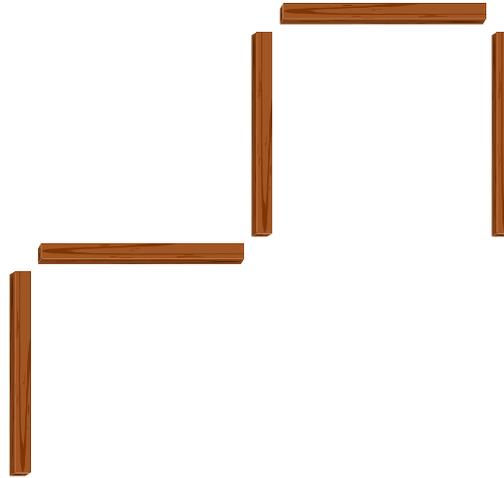
- https://de.wikipedia.org/wiki/Kürzester_Pfad
- <https://de.wikipedia.org/wiki/Breitensuche>



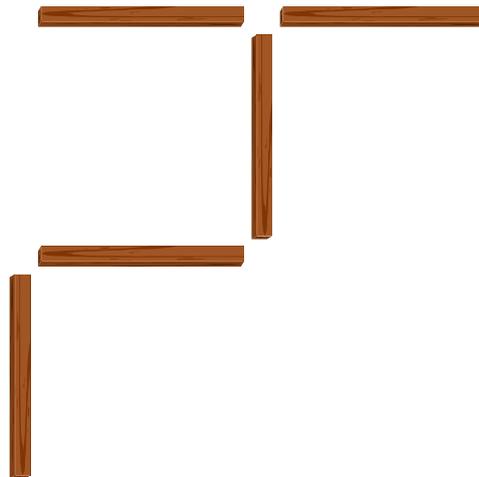


10. Fünf Hölzli

Fünf Hölzli liegen so auf dem Tisch:

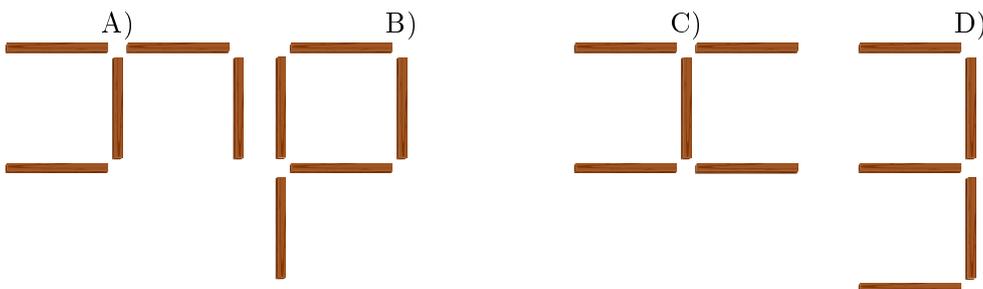


Nola nimmt ein Hölzli und legt es woanders hin. Jetzt liegen die Hölzli so:



Danach nimmt Bert ein Hölzli und legt es woanders hin.

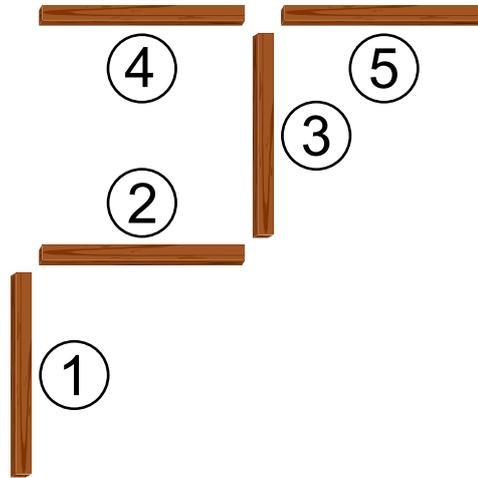
Wie können die Hölzli jetzt nicht liegen?





Lösung

Zum Erklären der Lösung werden die Hölzli durchnummeriert. Nach Nola liegen die Hölzli so:



Damit die Hölzli so wie in A) liegen, muss Bert das Hölzli 1 woanders hin legen.
Damit die Hölzli so wie in B) liegen, muss Bert das Hölzli 5 woanders hin legen.
Damit die Hölzli so wie in C) liegen, muss Bert das Hölzli 1 woanders hin legen.
Damit die Hölzli so wie in D) liegen, müsste Bert zwei Hölzli, nämlich die Hölzli 1 und 5 woanders hin legen. Er hat aber nur ein Hölzli woanders hin gelegt.

Dies ist Informatik!

Um die Lage der Hölzchen zu verändern, haben Nola und Bert eine einfache Möglichkeit, nämlich genau ein Hölzchen woanders hin zu legen. Wenn ein Freund ihnen sagen möchte, wie sie vorgehen sollen, kann er eine entsprechende Anweisung benutzen: „Lege ein Hölzchen woanders hin!“ Diese Anweisung ist jedoch nicht eindeutig. Er muss auch noch sagen, welches Hölzchen wohin muss. Wer Computer programmiert, braucht dazu eindeutige Anweisungen. Gewisse Anweisungen kennt der Computer bereits, neue Anweisungen können aus bereits bestehenden zusammengesetzt werden. Du kannst neue komplexeren Anweisungen erschaffen, indem du verschiedene Anweisungen hintereinander aufschreibst oder indem du aufschreibst, dass eine Anweisung mehrfach hintereinander ausgeführt werden soll. Du kannst auch festlegen, dass eine Anweisung nur unter bestimmten Bedingungen ausgeführt werden soll. Das sind die drei wichtigsten Möglichkeiten, in Computerprogrammen aus (anfänglich einfachen) Anweisungen immer komplexere Anweisungen zu machen.

Webseiten und Stichwörter

Anweisungen, Zustandsveränderung, Programmieren

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



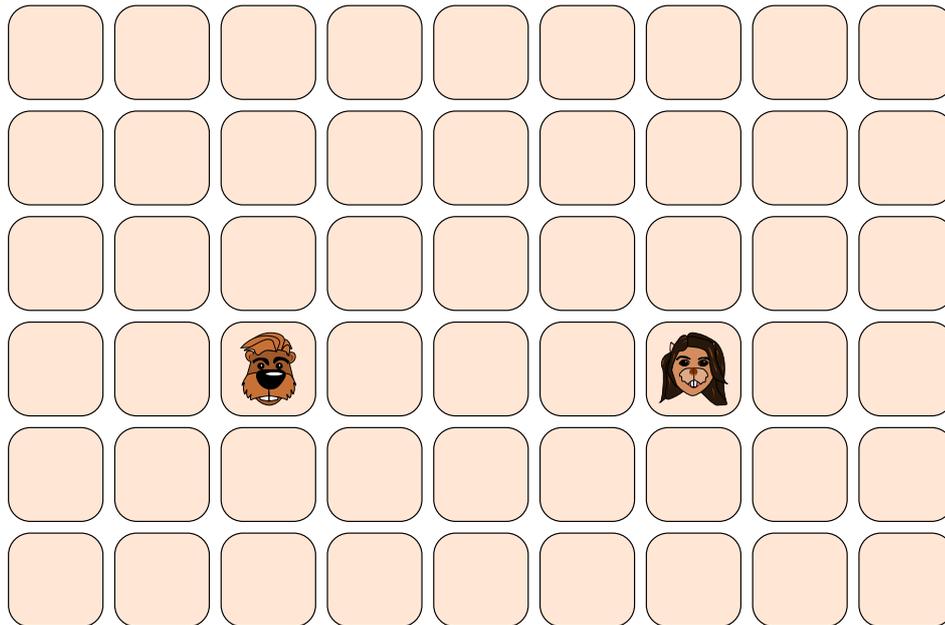
11. Dancing like the Stars

Ein Erdhörnchen (🐿️) und ein Biber (🦫) nehmen an einer Tanzshow teil. Je nach Reaktion der Zuschauermenge vollführen sie bestimmte Tanzschritte. Die folgende Tabelle zeigt, wie sie sich bewegen:

	Wow!	Kreisch!	Klatsch!	Buh!
🐿️	← ↑	↑ ←	← ← ↑	↓ ↓
🦫	↑ →	→ ↓	↑ ↑ ↑	← ←

Wenn zum Beispiel die Zuschauermenge kreischt, wird das Erdhörnchen sich ein Feld nach oben und ein Feld nach links bewegen; gleichzeitig wird der Biber sich ein Feld nach rechts und ein Feld nach unten bewegen.

Sie starten auf der Tanzfläche an folgenden Positionen:



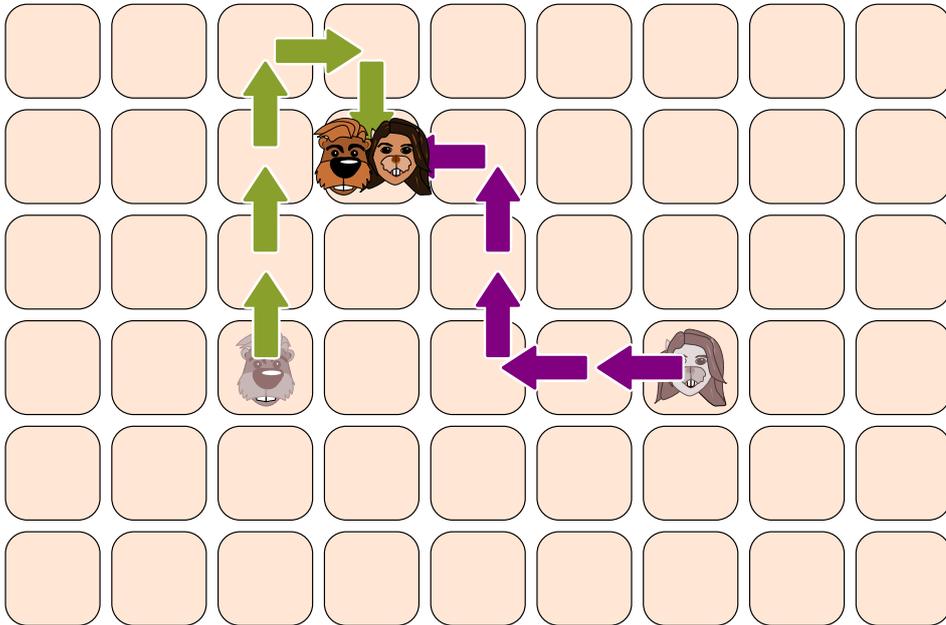
Welcher der folgenden Aktionen der Zuschauer lassen das Erdhörnchen und den Biber am Ende auf einem gemeinsamen Feld landen?

- A) Buh! Kreisch!
- B) Wow! Kreisch!
- C) Kreisch! Kreisch!
- D) Klatsch! Kreisch!

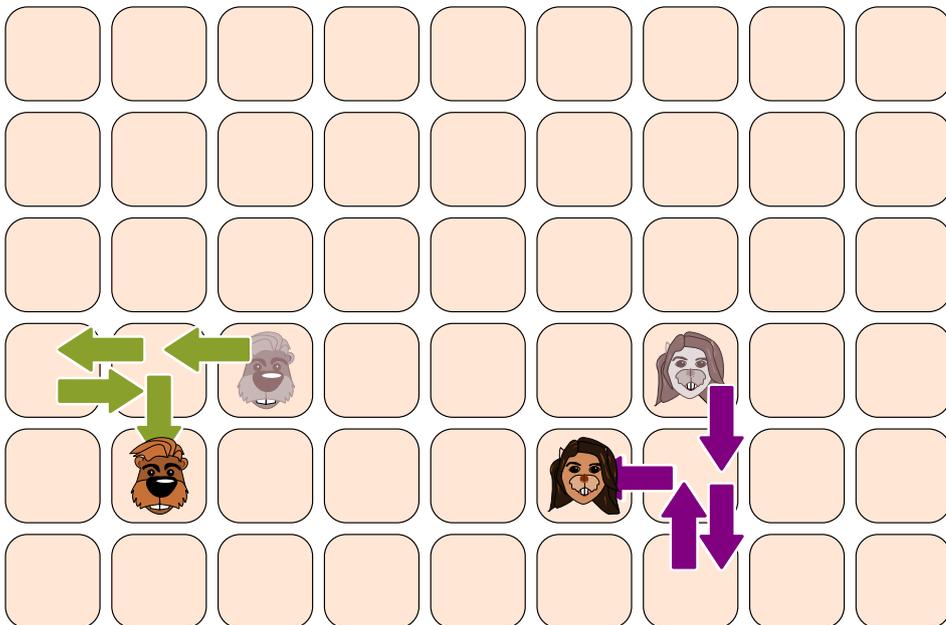


Lösung

Die Antwort D) „Klatsch! Kreisch!“ ist korrekt:

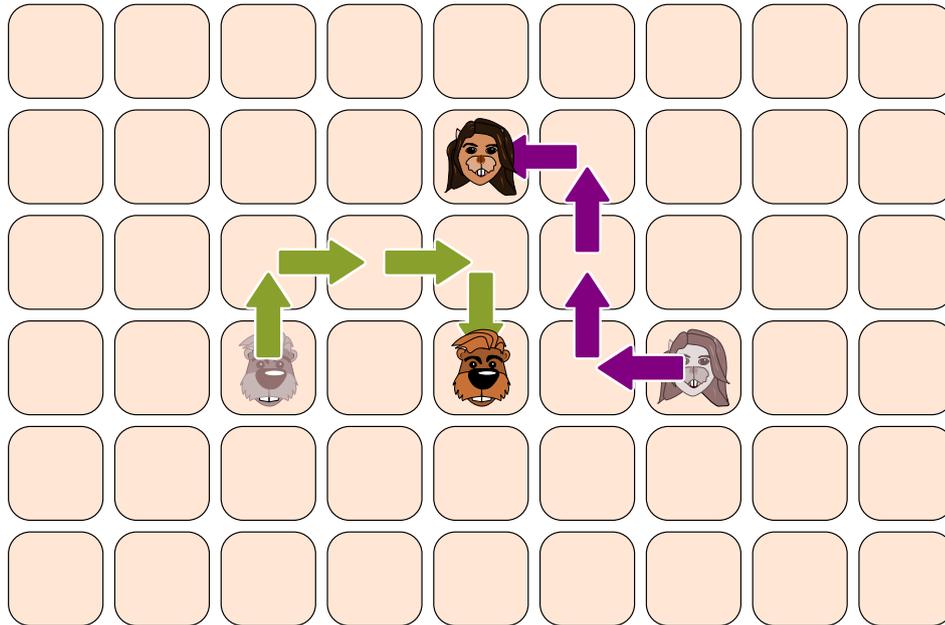


In allen anderen Fällen landen das Erdhörnchen und der Biber an einer anderen Stelle:
A) „Buh! Kreisch!“:

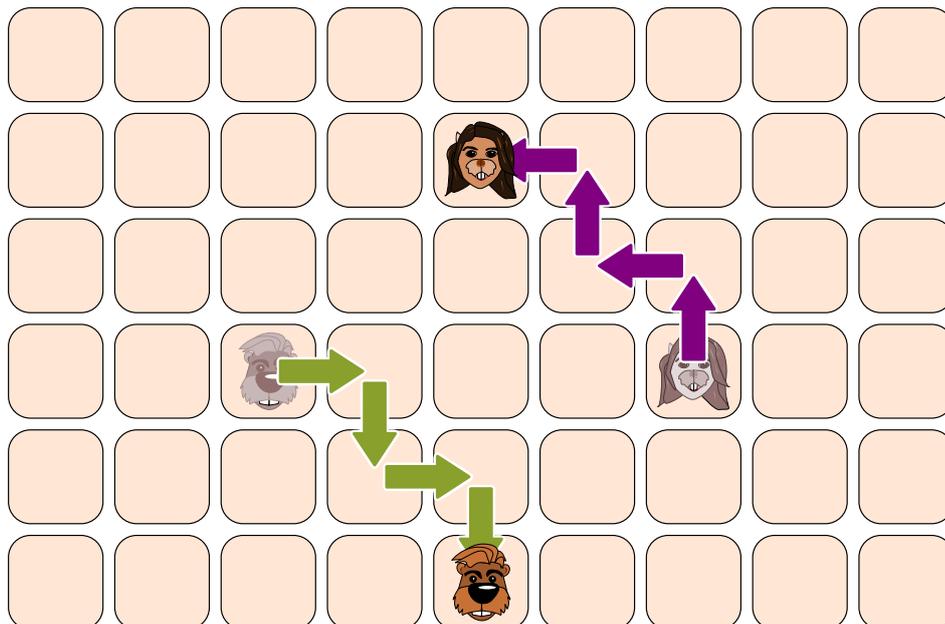




B) „Wow! Kreisch!“:



C) „Kreisch! Kreisch!“:



Dies ist Informatik!

Diese Aufgabe ist ein Beispiel für die Arbeit eines *Parallelrechners*. In diesem Fall handeln zwei *Akteure* unabhängig voneinander, jedoch nach vorher definierten Regeln. Wenn zwei Akteure kollidieren, also zur gleichen Zeit auf dieselbe Ressource zugreifen wollen (wie beispielsweise den Arbeitsspeicher oder ein angeschlossenes Gerät), muss geregelt werden, wer die Ressource zuerst nutzt. Dies kann man beispielsweise durch *Semaphoren* regeln. Dies gibt einem Akteur die Möglichkeit, eine Ressource zu reservieren, zu nutzen und danach wieder freizugeben. Während der Reservierung darf kein anderer Akteur darauf zugreifen. Damit eine Ressource nicht gleichzeitig reserviert wird, gibt es in



der Regel ein zentrales System (z.B. das Betriebssystem), das die Reservierungen der Ressourcen verwaltet und Anfragen in einer eindeutigen Reihenfolge abarbeitet.

Als Du Dir die Abfolge der Tanzschritte auf dem Bildschirm überlegt hast, hast Du übrigens eine *Simulation* der zwei Akteure vorgenommen. Computersimulationen helfen an sehr vielen Stellen in der Wirklichkeit, zum Beispiel werden Wetterprognosen mit Hilfe von Computersimulationen erstellt.

Webseiten und Stichwörter

Parallelrechner, Semaphor, Simulation

- <https://de.wikipedia.org/wiki/Parallelrechner>
- [https://de.wikipedia.org/wiki/Semaphor_\(Informatik\)](https://de.wikipedia.org/wiki/Semaphor_(Informatik))
- <https://de.wikipedia.org/wiki/Computersimulation>



12. Japanischer Ehrenname

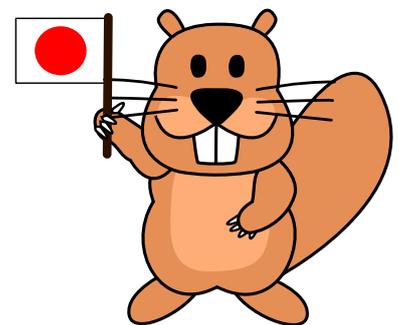
Eine Freundin aus Japan erzählt davon, dass nach einer alten japanischen Tradition aus den Buchstaben des Vornamens Ehrennamen gemacht werden, indem jeder Buchstabe mit einer Silbe ersetzt wird.

A → ka	F → lu	K → me	P → mor	U → do	Z → zi
B → pi	G → ji	L → ta	Q → ke	V → ru	
C → mi	H → ri	M → rin	R → shi	W → mei	
D → te	I → ki	N → to	S → ari	X → na	
E → ku	J → zu	O → mo	T → chi	Y → fu	

Ein Freund von ihr aus Kroatien wird zum Beispiel „Zukame Moru“ genannt.

Wie heisst ihr Freund wirklich?

- A) Josip
- B) Jani
- C) Jakov
- D) Jurica





Lösung

Ihr Freund heisst Jakob. Man kann vorne anfangen und nach der Silbe „Zu“ suchen („J“). Nach der zweiten Silbe „ka“ („A“) fallen die Antwortmöglichkeiten Josip und Jurica bereits weg und nach der dritten Silbe „me“ („K“) kann es eigentlich nur noch die Antwort Jakob sein. Man kann „Moru“ aber leicht überprüfen, indem man „O“ („mo“) und „V“ („ru“) übersetzt. Das geht schneller als nach „mo“ und „ru“ zu suchen, weil die Tabelle nach den Buchstaben sortiert ist, während die Silben unsortiert sind.

Ganz schnell kann man die Lösung übrigens finden, indem man von Anfang an nach der letzten Silbe „ru“ sucht. Da diese für „V“ steht und nur Jakob mit „V“ endet, ist man eigentlich schon fertig.

Dies ist Informatik!

Du hast vielleicht schon früher von solchen Spielen gehört. Manchmal werden sie auch als Geheimsprachen verwendet. Alle basieren darauf, dass ein Buchstabe durch einen anderen Buchstaben oder eine eindeutige Silbe ersetzt werden. In der Informatik nennt man dies ein *Wortersetzungssystem* (auch wenn hier Buchstaben durch Silben ersetzt werden) oder *Semi-Thue-System*.

Sichere Geheimsprachen sind solche Systeme nicht. Sie gehören zu den *monoalphabetischen Verschlüsselungsalgorithmen* und sind sehr leicht zu knacken, sogar ohne Computer.

In „Moru“ steckt auch die Silbe „mor“, die den Buchstaben „P“ ersetzt. Die Silbe „mo“, die den Buchstaben „O“ ersetzt, ist ein sogenanntes *Präfix* von „mor“. Soll ein Computerprogramm den Ehrennamen entschlüsseln, muss der Programmierer sich überlegen, wie er mit Präfixen umgeht.

Die Silben in diesem Ersetzungssystem sind übrigens solche wie sie auch in Japan verwendet werden. Daher klingen alle übersetzten Namen so japanisch.

Webseiten und Stichwörter

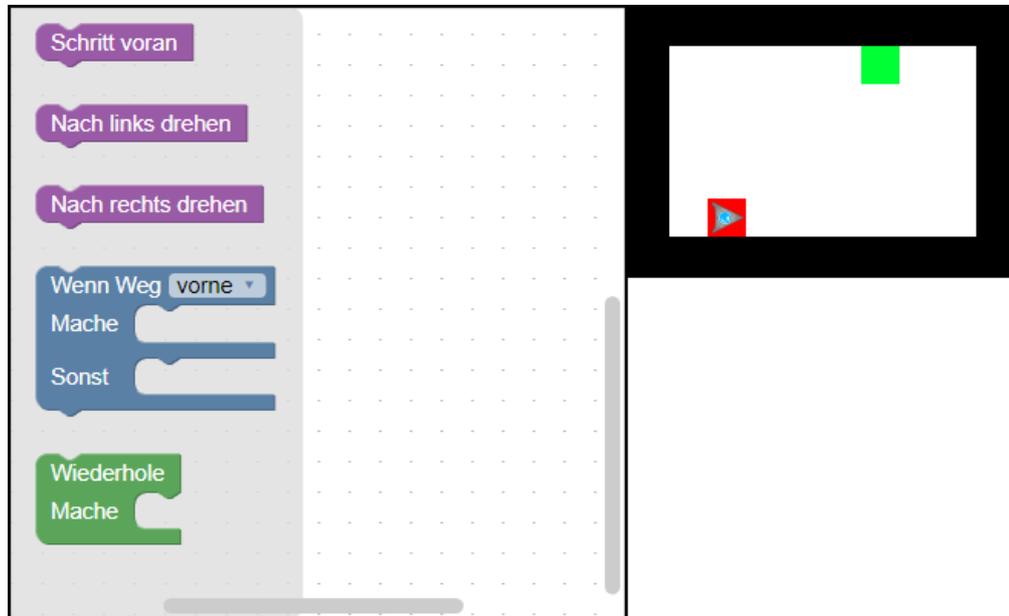
Wortersetzungssystem, Semi-Thue-System, Monoalphabetische Verschlüsselung

- <https://de.wikipedia.org/wiki/Semi-Thue-System>



13. Kurzes Programm

Ein dreieckiger Roboter soll vom dunkelroten Feld zum hellgrünen Feld gelangen. Er kann jedoch nur sehr kurze Programme verarbeiten.



Wähle dazu Programmbausteine und füge sie auf der Arbeitsfläche zusammen. Du darfst höchstens vier Bausteine für dein Programm benutzen.



Lösung

Dies wäre eine mögliche Lösung:



Die Idee ist, entlang der Wand zu fahren und sich in jeder Ecke nach links zu drehen. Dann kommt man automatisch zum hellgrünen Feld.

Dies ist Informatik!

In dem 8×5 Schritte grossen Raum führt der Roboter mindestens neun Bewegungen aus, um das Feld mit dem Kreis zu erreichen: vor – vor – vor – vor – links – vor – vor – vor – vor.

Das Programm funktioniert aber nur mit vier Bausteinen, daher braucht es eine Schleife, um beliebig viele Vorwärtsschritte und Linkswendungen zu vollführen, bis ein Ziel erreicht ist. Ausserdem macht es sich eine bedingte Anweisung zunutze, um bis zur Wand zu fahren und dann die Richtung zu ändern. So werden auch Kollisionen mit der Wand vermieden.

Allerdings findet dieses einfache Programm nicht den kürzesten Weg. Zudem funktioniert es nicht in jeder Situation. Würde das Ziel nicht an einer Wand liegen, würde der Roboter es nicht finden können.

Webseiten und Stichwörter

Programmieren, Programmschleife, Bedingung, Anweisung

- <https://en.wikipedia.org/wiki/Blockly>
- <http://primalogo.ch/>
- <http://www.abz.inf.ethz.ch/primarschulen-stufe-sek-1/programmieren-fur-kinder/>
- http://www.swisseduc.ch/informatik/programmiersprachen/scratch_werkstatt/
- <http://ilearnit.ch/de/2b/explain.html>
- <https://scratch.mit.edu/>
- <http://kinderlabor.ch/informatik-fuer-kinder/programmieren-mit-scratch/>

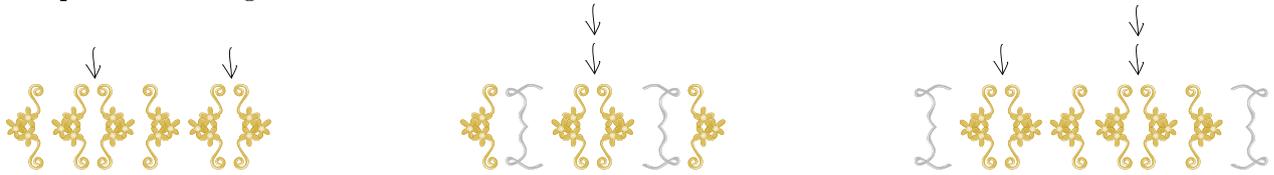


14. Klammeranhänger

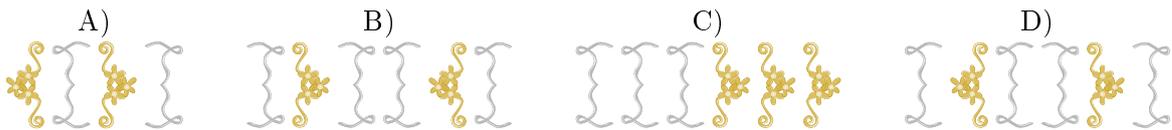
Familie Biber produziert selbstgemachte Anhänger für ein Mittelalterfest. Dazu verwenden sie klammernförmige Ornamente, die immer paarweise verwendet werden. Um einen Anhänger herzustellen, beginnt man mit einem der beiden folgenden Paare:



Danach werden wiederholt weitere Klammernpaare an einer beliebigen Stelle eingefügt, wie die drei Beispiele unten zeigen:



Welcher der folgenden Anhänger wurde mit der obigen Methode hergestellt?





Lösung

D) ist die richtige Antwort. In das ursprüngliche Klammernpaar wurde ein zusätzliches Klammernpaar eingefügt und in dieses wiederum ein weiteres.

Alle anderen Anhänger wurden nicht mit der beschriebenen Methode hergestellt:

- A) Von links beginnend ist der erste Fehler bei Position 3: eine Klammer wurde geschlossen, bevor die Klammer von Position 2 geschlossen wurde.
- B) Von links beginnend ist der erste Fehler schon bei Position 1: Eine Klammer wurde geschlossen, bevor sie geöffnet wurde
- C) Von links beginnend ist der erste Fehler bei Position 4: eine Klammer wurde geschlossen, bevor sie überhaupt geöffnet wurde.

Dies ist Informatik!

Die Regeln zur Herstellung von Anhängern sind genau die gleichen Regeln wie für Klammernausdrücke in der Mathematik oder in der Informatik. Ausdrücke ohne Fehler nennt man *wohlgeformt*. Einen wohlgeformten Ausdruck nennt man auch *syntaktisch korrekt* in dem Sinne, dass sie die vorgegebenen syntaktischen Regeln, der Grammatik einer formalen Sprache, befolgen. Syntaxfehler sind in der Regel viel leichter zu beheben als Fehler, die zwar syntaktisch korrekt sind aber andere, mitunter subtile, Denkfehler enthalten. Die letzteren nennt man *semantische Fehler*.

Webseiten und Stichwörter

Wohlgeformtheit, Syntax, Semantik

- <https://de.wikipedia.org/wiki/Syntax>
- <https://de.wikipedia.org/wiki/Semantik>



15. Schülerzeitung

Bei der Schülerzeitung arbeiten zehn Freiwillige mit. Immer freitags benutzen sie ihre Freistunden dazu. Der Stundenplan zeigt farbig an, wer wann Zeit hat.

	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00
Anna								
Bea								
Celine								
David								
Emma								
Flo								
Gioa								
Hans								
Ida								
Jakob								

Für ihre Arbeit an der Schülerzeitung sollen ihnen neue Notebooks von der Schule zur Verfügung gestellt werden.

Wie viele neue Notebooks muss die Schule mindestens bereitstellen, dass jede(r) Freiwillige an einem Notebook arbeiten kann?

- A) vier
- B) fünf
- C) sieben
- D) zehn



Lösung

Antwort A) ist falsch. Zwischen 9:00 Uhr und 10:00 Uhr jedoch müssten die Personen 1, 3, 5, 8 und 9 gleichzeitig arbeiten, dafür würden vier Notebooks nicht genügen.

Antwort B) ist richtig. Wenn eine bestimmte Anzahl von Freiwilligen gleichzeitig eine Freistunde haben, brauchen sie auch genau die gleiche Anzahl Notebooks. In den Stunden 9 Uhr bis 10 Uhr sowie 10 Uhr bis 11 Uhr wollen jeweils fünf Freiwillige arbeiten, in allen anderen Stunden weniger. Es werden also maximal fünf Notebooks benötigt.

	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00
Anna								
Bea								
Celine								
David								
Emma								
Flo								
Gioa								
Hans								
Ida								
Jakob								

Da fünf Notebooks genügen, sind die Antworten C) und D) falsch.

Dies ist Informatik!

Um Zusammenhänge in grossen Datenmengen zu verstehen, ist es hilfreich, eine geeignete Darstellung der Daten zu wählen, beispielsweise eine Tabelle, ein Diagramm oder ein Graph. Abhängig vom Ziel wird man unterschiedliche Darstellungen wählen.

Für diesen Task macht es Sinn, die Daten so darzustellen:

	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00
	Celine	Celine	Bea	Bea	Bea	Bea		
		Emma	Emma	Flo	Flo	Jakob	Jakob	
		Anna	Anna		David	David	David	
	Ida	Ida	Ida					
		Hans	Gioa	Gioa		Gioa	Gioa	

In dieser Darstellung wird den Notebooks zu bestimmten Uhrzeiten jeweils eine Person zugeordnet. So kann man schnell erkennen, dass maximal fünf Notebooks gebraucht werden.

Solche Planungen braucht es häufig, auch sollen die vorhandenen Ressourcen möglichst optimal genutzt werden. Jedes Hotel hat beispielsweise ein System zum Reservieren von Räumen. Dabei dürfen sich die Reservierungen nicht überschneiden, zudem wäre es natürlich praktisch, wenn die Räume möglichst ausgebucht sind. Dasselbe gilt für Klassenzimmer in einer Schule, auch hier ist ein optimaler Plan hilfreich, auch wenn „optimal“ für verschiedene Gruppen etwas anderes bedeuten kann: Möchte man beispielsweise eine Klasse möglichst immer in demselben Zimmer unterrichten, oder möchte man, dass die Lehrpersonen möglichst wenig das Zimmer wechseln?

Eine andere Darstellungsmethode für einen solchen Plan wäre ein Intervallgraph, bei dem die Knoten jeweils ein Intervall darstellen und die Kanten jeweils eine Überlappung der Intervalle bedeuten.



Webseiten und Stichwörter

Planung, Intervallgraph

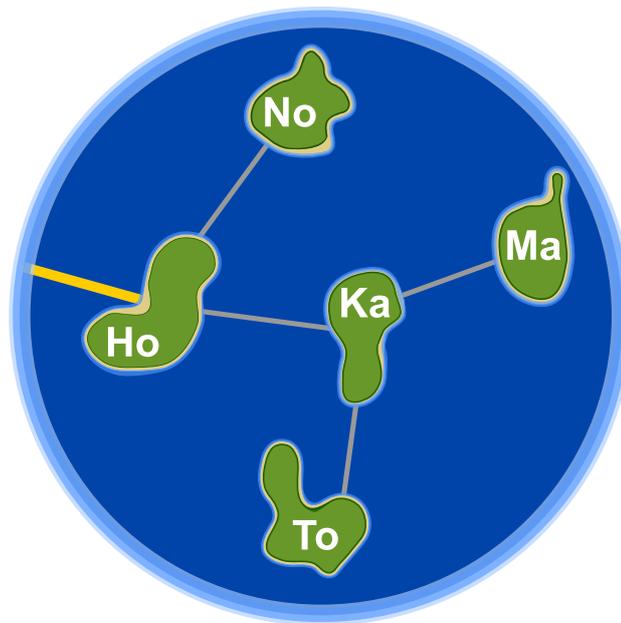
- <https://de.wikipedia.org/wiki/Intervallgraph>





16. Honomakato

Die Inselgruppe Honomakato besteht aus den fünf Inseln Ho, No, Ma, Ka und To. Die Hauptinsel Ho ist über ein Kabel mit dem Internet verbunden. Ausserdem sind einige Kabel verlegt: zwischen Ho und No, Ho und Ka, Ka und Ma sowie Ka und To. So sind alle Inseln mit Ho verbunden und dadurch auch mit dem Internet.



Die Bewohner von Honomakato möchten eine zuverlässige Verbindung aller Inseln mit dem Internet: Auch wenn irgendeines der Kabel zwischen den Inseln beschädigt würde, soll jede Insel noch mit dem Internet verbunden sein.

Sorge dafür, dass Honomakato eine zuverlässige Verbindung mit dem Internet bekommt. Verlege dazu zwei weitere Kabel zwischen den Inseln. Es gibt mehrere richtige Antworten.

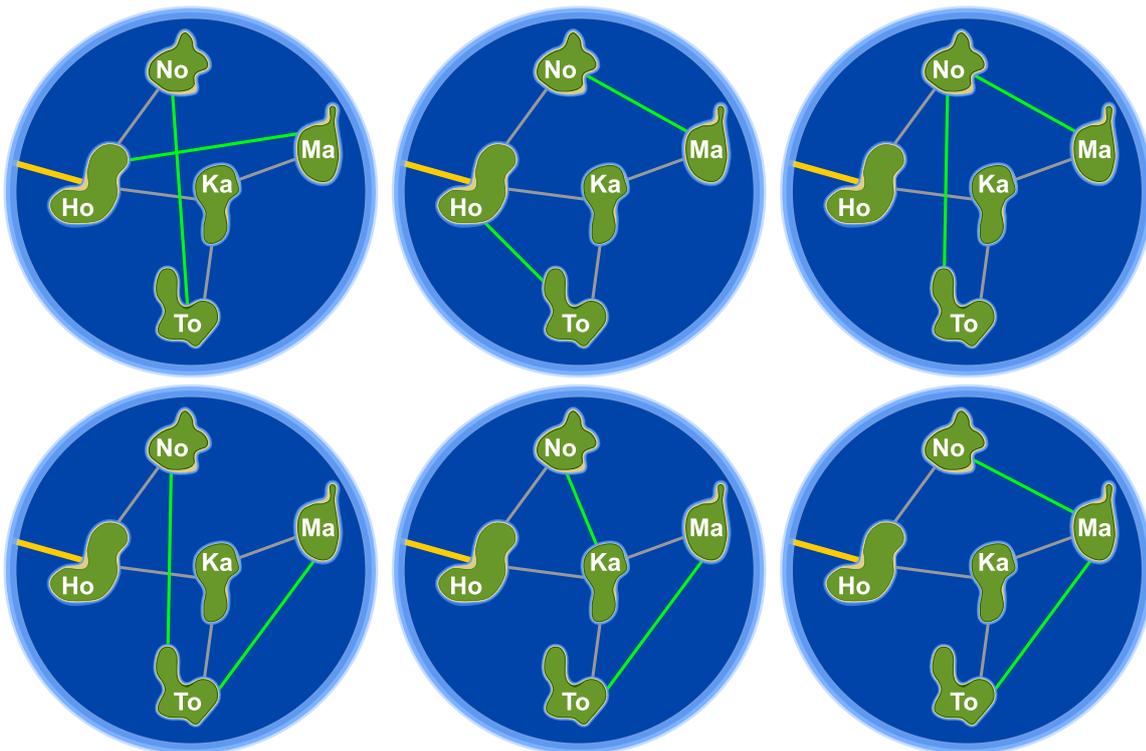


Lösung

Durch das Verlegen zweier weiterer Kabel kann dafür gesorgt werden, dass Honomakato eine zuverlässige Verbindung mit dem Internet bekommt. Dabei gibt es sechs verschiedene Möglichkeiten. Die einzelnen Kabel schützen jeweils bestimmte Inseln davor, durch die Beschädigung eines Kabels nicht mehr mit dem Internet verbunden zu sein.

1. Ho-Ma und No-To: Ho-Ma schützt Ma und Ka, No-To schützt No und To.
2. Ho-To und No-Ma: Ho-To schützt To und Ka, No-Ma schützt Ma, No, und Ka.
3. No-To und No-Ma: No-To schützt No, To und Ka, No-Ma schützt Ma, No, und Ka.
4. No-To und Ma-To: No-To schützt No, To und Ka, Ma-To schützt Ma und To.
5. No-Ka und Ma-To: No-Ka schützt No und Ka, Ma-To schützt Ma und To.
6. No-Ma und Ma-To: No-Ma schützt Ma, No, und Ka, Ma-To schützt Ma und To.

Für die Lösungen gilt jeweils: Jede Insel ist mit mindestens zwei Verbindungen ausgestattet, und Honomakato lässt sich nicht in zwei Teile gruppieren, zwischen denen nur eine Verbindung existiert.



Dies ist Informatik!

Das Kabelnetzwerk, mit dem die Inseln von Honomakato mit dem Internet verbunden wurden, ist einerseits ein kleiner Teil des Internet. Andererseits ist sein Aufbau aber auch ein Beispiel für den Aufbau des gesamten Internet. Die Router, Server und anderen Geräte mit fester Internetadresse sind Knoten im grossen Netzwerk „Internet“, genauso wie hier die Inseln im Netzwerk von Honomakato. Das Internet wurde in den 1960er Jahren als robustes Netzwerk erfunden. Der Ausfall von Verbindungen zwischen einzelnen Netzwerkknoten sollte nicht zum Ausfall des gesamten Netzwerks führen.



Daher verbindet man die Knoten mehrfach miteinander und konfiguriert sie so, dass beim Ausfall oder der Überlastung einer Leitung einfach eine andere Leitung verwendet wird. Auch für andere Netzwerkarten wie Verkehrsnetze oder Versorgungsnetzwerke ist es wichtig, dass es keine einzelne Verbindung oder auch einzelnen Knoten gibt, deren Ausfall ein komplettes Netzwerk ausser Gefecht setzt.

Die Informatik setzt Graphentheorie ein, um Berechnungen über Netzwerke durchzuführen. Ein Graph ist (theoretisch) ein Netzwerk aus Knoten und Verbindungen (Kanten genannt) zwischen Knoten. Ein Graph heisst „zusammenhängend“, wenn für jedes Paar von Knoten A und B gilt, dass B mit A durch einen Weg über eine oder mehrere Kanten verbunden ist. Eine Kante in einem Graphen, die vorhanden sein muss, damit der Graph zusammenhängend ist, wird als Brücke bezeichnet. Die Informatik kennt Verfahren, um Brücken in Graphen zu finden. Von Robert Tarjan stammt ein solcher (effizienter) Algorithmus – sowie viele andere Algorithmen auf Graphen.

Webseiten und Stichwörter

Dynamische Datenstruktur, Graph, Brücke

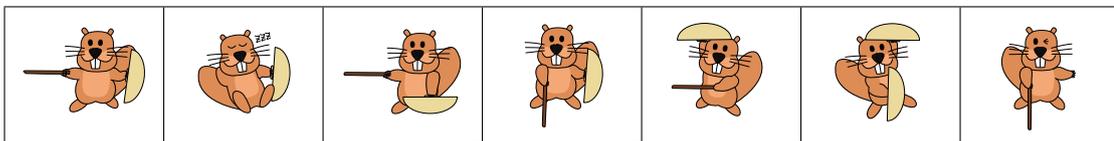
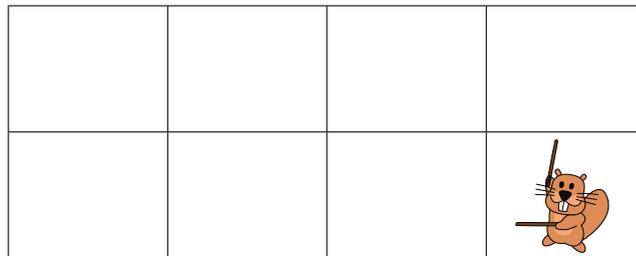
- [https://de.wikipedia.org/wiki/Trenner_\(Graphentheorie\)](https://de.wikipedia.org/wiki/Trenner_(Graphentheorie))





17. Japanischer Stockkampf

Lucia und ihre Freunde sind Mitglieder eines Clubs für japanischen Stockkampf. Für ein Foto möchten sie sich auf dem Schulhof so aufstellen, dass jeder Stock auf ein Schild zeigt. Dafür wurden Felder auf den Schulhof gezeichnet. Lucia hat sich bereits in Pose gestellt, darunter sind die Lieblingsposen ihrer Freunde:

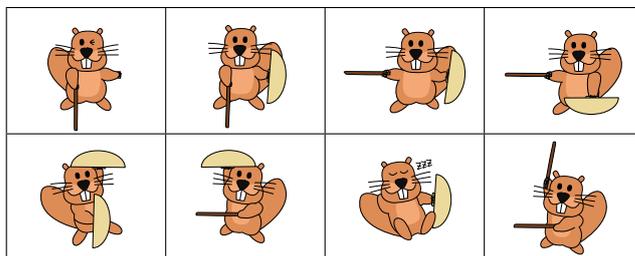


Ordne die Bilder der Freunde den Feldern auf dem Schulhof zu, so dass am Ende jeder Stock auf ein Schild zeigt.



Lösung

So ist es richtig:



Die Bilder der Freunde müssen so in die Felder bewegt werden, wie es im Bild oben gezeigt wird. Dann zeigt jeder Stock auf ein Schild. Eine andere Möglichkeit, die Bilder so anzuordnen, dass diese Bedingung erfüllt ist, gibt es nicht.

Dies ist Informatik!

Sieben Bilder müssen an die richtige Stelle geschoben werden. Wer versucht diese Aufgabe durch einfaches Ausprobieren zu lösen braucht viel Zeit. Denn es gibt $1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 = 7! = 5040$ verschiedene Möglichkeiten, die sieben Bilder zu arrangieren. Die meisten davon sind natürlich falsch. Mit ein bisschen Logik findest du die Lösung schneller. Überlegen wir mal:

1. Alle Biber, die einen Stock oder ein Schild (es gibt gleich viele Stöcke und Schilde) nach oben halten, müssen in der unteren Reihe stehen.
2. Alle Biber, die einen Stock oder ein Schild nach unten halten, müssen in der oberen Reihe stehen.
3. Es gibt nur einen einzigen Biber, der sein Schild nach unten hält und deshalb oberhalb von Lucia stehen kann.

Mit diesen Regeln kannst du den Suchraum für das Finden einer richtigen Lösung auf wenige Möglichkeiten eingrenzen. Ein Verfahren zum systematischem Ausprobieren aller Lösungsmöglichkeiten nach dem Prinzip von „Versuch und Irrtum“ ist das *Backtracking*. Ein solches Verfahren ist aber nur dann schnell genug, wenn der Suchraum klein ist. Deshalb ist das Eingrenzen durch logische Regeln so wichtig.

Webseiten und Stichwörter

Logisches Denken, Schlussfolgerung

- <https://de.wikipedia.org/wiki/Backtracking>
- http://www.inf-schule.de/grenzen/komplexitaet/affenpuzzle/einstieg_affenpuzzle



18. Opas Marmelade

Weil ihr Opa Marmelade kocht, helfen Anna, Peter und Lisa, die Marmelade in Gläser abzufüllen. Dazu müssen sie jeweils diese Arbeitsschritte erledigen – und zwar genau in dieser Reihenfolge:



Ein Glas spülen dauert 3 Minuten.



Marmelade in ein Glas füllen dauert 2 Minuten.



Ein Glas zu verschliessen dauert 1 Minute.

Anna, Peter und Lisa wollen sich die Arbeit aufteilen und dazu einen Plan erstellen. Dabei müssen sie beachten: Ein Arbeitsschritt muss komplett erledigt sein, bevor der nächste drankommen kann. Ein Glas muss ganz sauber sein, bevor man Marmelade einfüllt. Und das Glas darf natürlich erst geschlossen werden, wenn es fertig gefüllt ist.

Der folgende Plan ist also nicht möglich:

ANNA										
PETER										
LISA										

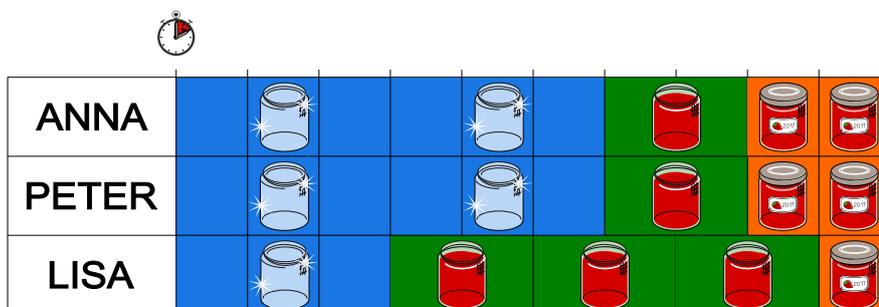
Anna, Peter und Lisa wollen in 10 Minuten möglichst viele Gläser abfüllen. Erstelle dazu einen Plan:

ANNA										
PETER										
LISA										



Lösung

Hier ist ein Plan, mit dem Anna, Peter und Lisa fünf Gläser in 10 Minuten abfüllen können. Es gibt noch andere Pläne, mit denen sie das ebenfalls schaffen können. In diesen Plänen sind die drei Zeilen – also die Folgen von Arbeitsschritten, die die einzelnen Kinder erledigen sollen – gleich wie im abgebildeten Plan, aber anderen Kindern zugeordnet.



Mehr als fünf Gläser können die Kinder in 10 Minuten nicht abfüllen. Das Abfüllen eines Marmeladenglases dauert 6 Minuten; denn $3 + 2 + 1 = 6$. Jedes der drei Kinder hat 10 Minuten Arbeitszeit. Somit beträgt die gesamte Arbeitszeit 30 Minuten. Es ist nicht möglich in 10 Minuten mehr als 5 Gläser fertig zu stellen; denn $30 : 6 = 5$. Fünf Gläser sind also das theoretische Optimum. Es kann mit dem abgebildeten Plan erreicht werden.

Dies ist Informatik!

In dieser Biberaufgabe arbeiten Anna, Peter und Lisa gleichzeitig, also parallel an den einzelnen Schritten eines Arbeitsvorgangs. Parallelverarbeitung gibt es auch in der Informatik. Zum Beispiel werden in Rechenzentren grosse Datenmengen von vielen Computern zur gleichen Zeit (parallel) verarbeitet. Dabei müssen – wie in der Aufgabe – einzelne Arbeitsschritte auf die Computer in sinnvoller Weise verteilt werden. Aber auch in einem einzigen Computer oder einem Smartphone gibt es mehrere Arbeitseinheiten (Prozessorkerne), die Arbeitsschritte parallel verarbeiten können. Die Erstellung eines Arbeitsplans (mit dem natürlich möglichst viel erledigt werden soll) nennt die Informatik auch „*Scheduling*“.

Aber auch ausserhalb der Informatik gibt es ähnliche Probleme. Für die Planung von Abläufen in einem grossen Projekt verwendet man die Netzplantechnik. In einem Netzplan ist festgelegt, in welcher Reihenfolge Arbeitsschritte ausgeführt werden und wie lange sie dauern. Die Aufstellung eines guten Netzplans ist wie Scheduling. Informatikkenntnisse helfen also auch, wenn man gar nicht als Informatikerin oder Informatiker arbeitet.

Webseiten und Stichwörter

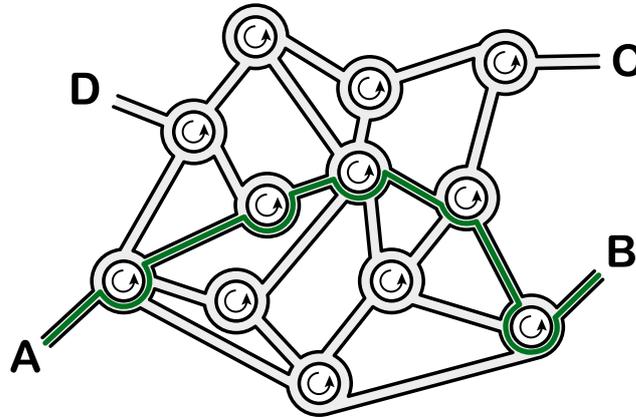
Scheduling, Parallelverarbeitung

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



Lösung

Sie führt zum Punkt B:



Dies ist Informatik!

Diese Aufgabe ist ein gutes Beispiel für formalisierte Information. Ein Steuerungscomputer kann mit einer ausformulierten Wegbeschreibung nicht viel anfangen. Wenn man aber eine Wegbeschreibung wie im Beispiel als Folge von Zahlen angeben kann, dann ist die Information in einer Form, die ein Computer als Folge von Befehlen interpretieren kann. So eine Befehlsfolge nennt man auch Sequenz und ist ein wichtiger Baustein in fast allen Programmiersprachen.

In diesem Beispiel ist es hilfreich, dass das Strassennetz gleichförmig aufgebaut ist: Alle Kreuzungen sind Kreisel. Solche gleichförmigen Strukturen nennt man homogen, im Gegensatz zu gemischten Strukturen, die als heterogen bezeichnet werden. Homogene Strukturen sind für Programme leichter abzarbeiten als heterogene, weshalb man sie in der Informatik bevorzugt, wenn es keinen zwingenden Grund dagegen gibt.

Webseiten und Stichwörter

Sequenzen, Ausführen von Programmen, Formale Sprache

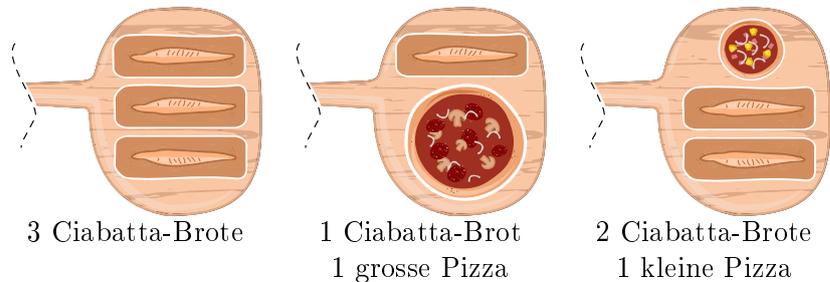
- https://de.wikipedia.org/wiki/Imperative_Programmierung



20. Pizzeria Biberia

Die Pizzeria Biberia hat nur einen Pizzaofen, daher können nur wenige Gerichte gleichzeitig gebacken werden.

Die Kombinationen sind:



Die Gerichte brauchen unterschiedlich lang: Eine kleine Pizza muss 10 Minuten im Ofen sein, eine grosse Pizza muss 15 Minuten im Ofen sein und ein Ciabatta-Brot muss 20 Minuten im Ofen sein. Der Pizzabäcker kann jedoch Gerichte einzeln in den Ofen legen oder aus dem Ofen holen.

Heute ist viel los: Es werden eine kleine Pizza, zwei grosse Pizzen und vier Ciabatta-Brote bestellt. Die Gäste sind hungrig und möchten ihre Bestellung so schnell wie möglich bekommen.

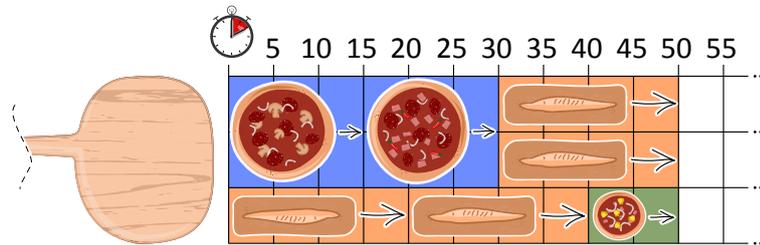
In wie vielen Minuten kann der Pizzabäcker die Bestellung schnellstmöglich backen lassen?





Lösung

Es gibt mehrere optimale Lösungen. Am einfachsten findet man eine dieser Lösungen, wenn man die beiden grossen Pizzen hintereinander backt, parallel dazu zwei Ciabatta-Brote und dann mit zwei Ciabatta-Broten und einer kleinen Pizza ergänzt:



Diese Lösung benötigt 50 Minuten. Eine schnellere Lösung kann es nicht geben, da der Platz im Ofen während der 50 Minuten immer voll ausgenutzt wird.

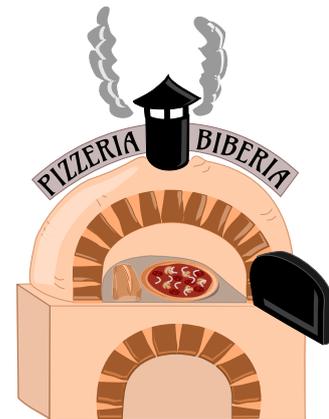
Dies ist Informatik!

Wenn man versucht, einen Ablaufplan (engl. *schedule*) für eine beschränkte Ressource (in diesem Fall der Pizzaofen) zu erstellen, möchte man in der Regel eine optimale Lösung finden, beispielsweise bezüglich der Kosten oder (wie in diesem Fall) bezüglich der Zeit. So wird die Ressource optimal genutzt und damit auch die kürzeste Backzeit erreicht.

Ein weit verbreitetes Verfahren zum Finden von optimalen Ablaufplänen ist das Rundlauf-Verfahren (engl. *Round-Robin*). Bei diesem Verfahren sind alle Prozesse, die stattfinden sollen, in einer Warteschlange. Jeder Prozess bekommt dann für eine kurze Zeit die Ressource zur Verfügung gestellt. Wenn er am Ende der kurzen Zeit noch nicht fertig ist, wird er wieder neu in die Warteschlange eingereiht.

Für diesen Pizzaofen jedoch macht die Strategie nicht viel Sinn ... man nimmt ja keine halbfertige Pizza aus dem Ofen und schiebt sie später wieder rein. Daher kann man in diesem Fall mit einer gierigen (engl. *greedy*) Strategie vorgehen: man fängt mit dem grössten Prozess an (einer grossen Pizza, da diese am meisten Platz wegnimmt) und versucht dann, mit nächstkleineren Prozessen aufzufüllen und so weiter. Da der Ofen (mit Einschränkungen) mehrere Dinge gleichzeitig backen kann, kann man auch immer jeweils das fertige Gericht herausnehmen und durch das nächste zu backende Gericht ersetzen.

Hat man aber am Ende die optimale Lösung gefunden? Das ist nicht offensichtlich, man könnte ja vielleicht eine bessere Lösung finden, indem man beispielsweise mit drei Ciabatta-Broten anfängt. Wenn man es aber geschafft hat, alle Ressourcen immer voll zu belegen (also in diesem Fall den Ofen immer voll zu haben), kann man sich sicher sein, dass es keine bessere Lösung gibt.



Webseiten und Stichwörter

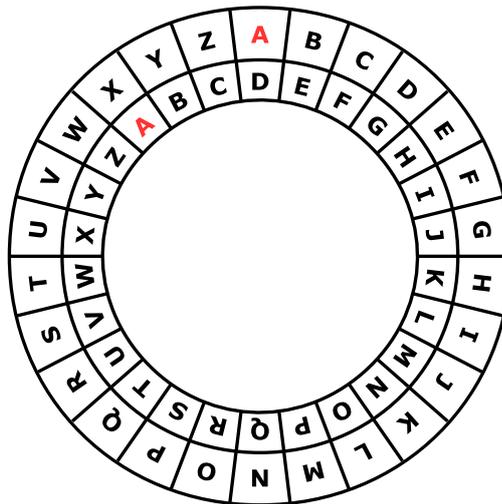
Schedule, Warteschlange, Ressource, Round-Robin, Greedy

- <https://de.wikipedia.org/wiki/Prozess-Scheduler>
- [https://de.wikipedia.org/wiki/Round_Robin_\(Informatik\)](https://de.wikipedia.org/wiki/Round_Robin_(Informatik))
- <https://de.wikipedia.org/wiki/Greedy-Algorithmus>



21. Geheime Bestellung

Anna bestellt im Restaurant mit geheimen Botschaften; nur Cäsar, der Koch, soll sie verstehen. Sie benutzt dazu ihre Geheimscheibe. Diese hat einen äusseren und einen inneren Ring mit den Buchstaben des Alphabets. Zu Anfang stehen die Ringe gleich: A (innen) passt zu A (ausseren), B passt zu B usw.



So erstellt Anna eine geheime Botschaft: Zuerst schreibt sie ihre Bestellung auf, z.B. PIZZA. Dann macht sie für jeden Buchstaben Folgendes:

1. Unter den Buchstaben schreibt sie eine „Dreh-Zahl,“.
2. Sie stellt den inneren Ring auf Anfang und dreht ihn dann um so viele Buchstaben gegen den Uhrzeigersinn, wie die Dreh-Zahl angibt.
3. In die Botschaft schreibt sie den Buchstaben, der nun zum Buchstaben aus der Bestellung passt.

Wenn Anna zum Beispiel PIZZA bestellen möchte und die Dreh-Zahlen 3, 1, 4, 1 und 5 benutzt, erstellt sie die geheime Botschaft SJDAF.

Bestellung	P	I	Z	Z	A
Dreh-Zahl	3	1	4	1	5
Geheime Botschaft	S	J	D	A	F

Für eine andere Bestellung hat Anna mit den Dreh-Zahlen 3, 1, 4, 1, 5, 9 und 2 die Botschaft OBWBLWC erstellt.

Wie lautet die Bestellung?

Bestellung							
Dreh-Zahl	3	1	4	1	5	9	2
Geheime Botschaft	O	B	W	B	L	W	C



Lösung

Die richtige Antwort ist LASAGNA:

Bestellung	L	A	S	A	G	N	A
Dreh-Zahl	3	1	4	1	5	9	2
Geheime Botschaft	O	B	W	B	L	W	C

Die Bestellung bekommt man mit Hilfe der Geheimscheibe heraus, indem man für jeden Buchstaben der Botschaft den inneren Ring gemäss der Dreh-Zahl nach links dreht. Nun sucht man den Buchstaben der Botschaft auf dem inneren Ring. Der Buchstabe der Bestellung ist dann der passende Buchstabe auf dem äusseren Ring.

Dies ist Informatik!

Anna verschlüsselt ihre Bestellungen, damit nur ihr Lieblings-Koch sie versteht. Verschlüsselung ist eines der ältesten Anliegen der Menschheit. Schon immer hat es Gründe dafür gegeben, Botschaften so zu übermitteln, dass nur die gewünschten Empfänger sie verstehen können. Es gibt viele verschiedene Verschlüsselungsverfahren. Aber immer gehören zwei Algorithmen dazu, nämlich einer zum Verschlüsseln und einer zum Entschlüsseln und beide benötigen für ihre Arbeit den zur Botschaft gehörigen Schlüssel.

Eines der einfachsten Verschlüsselungsverfahren geht auf Julius Cäsar zurück: Hier ist der Schlüssel eine Zahl, die eine Verschiebung im Alphabet angibt. Der Schlüssel 3 bedeutet z.B., dass der Buchstabe A einer Nachricht mit D verschlüsselt wird und B mit E usw. – und dass der Buchstabe D als A zu entschlüsseln ist, E als B usw. Beim Verschlüsseln und Entschlüsseln nach dieser Methode hilft die „Cäsar-Scheibe“, die in dieser Biberaufgabe beschrieben ist.

Verschlüsselungsverfahren, die für eine Botschaft nur einen Schlüssel verwenden, sind vergleichsweise unsicher. Das weiss Anna anscheinend, denn sie benutzt für jeden Buchstaben einen anderen Schlüssel – ganz ähnlich wie beim Verfahren von Vigenère. In diesem Verfahren wiederholen sich die Dreh-Zahlen bei längeren Botschaften; so wird der Schlüssel nicht zu lang. Aber auch dieses Verschlüsselungsverfahren ist bei längeren Botschaften letztlich unsicher.

Webseiten und Stichwörter

Kryptographie, Polyalphabetische Verschlüsselungsverfahren, Caesar-Verschlüsselung, Vigenère-Verschlüsselung

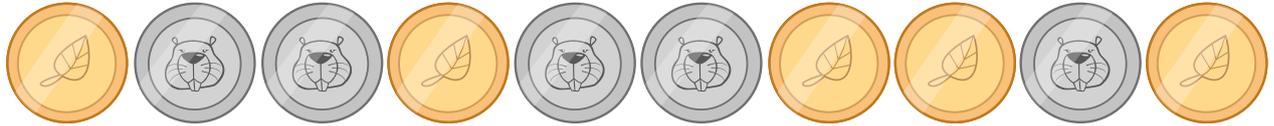
- <https://de.wikipedia.org/wiki/Caesar-Verschlüsselung>
- https://de.wikipedia.org/wiki/Polyalphabetische_Substitution



22. Münzdreh

Christine besitzt zehn Münzen, die auf der einen Seite golden (🟡) und auf der anderen Seite silbern (🟡) sind.

Sie legt die Münzen so auf den Tisch:



Wie häufig muss sie je zwei nebeneinanderliegende Münzen umdrehen, so dass am Ende alle goldfarbenen Seiten aufgedeckt sind?

- A) 1
- B) 2
- C) 4
- D) 6
- E) 8
- F) Es ist nicht möglich.



Lösung

Es ist nicht möglich.

Jedes Mal, wenn sie zwei Münzen umdreht, bleibt die Anzahl der silbernen Seiten gleich: Es werden zwei silberne Seiten mehr oder zwei silberne Seiten weniger. Die Parität, also ob die Anzahl der silbernen Seiten gerade oder ungerade ist, bleibt immer gleich.

Die Parität der silbernen Seiten ist am Anfang ungerade ... und sie bleibt immer ungerade. Eine Situation mit keiner silbernen Seite, was ja eine gerade Anzahl silberner Seiten wäre, kann also nie erreicht werden.

Dies ist Informatik!

Paritäten können schnell und einfach berechnet werden. Man kann mit ihnen einfach überprüfen, ob eine Übertragung korrekt war (wie beispielsweise ein an der Kasse eingelesener Barcode) oder eine Zahl richtig eingegeben wurde (wie beispielsweise eine Kontonummer im Online-Banking). Wenn man etwas komplizierte Berechnungen vornimmt, kann man sogar bestimmte Fehler korrigieren, ohne dass die Daten neu übertragen werden müssen.

Webseiten und Stichwörter

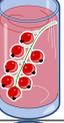
Parität, Paritätsbit

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



23. „Saftladen“

Auf dem Weg in den Urlaub halten vier Freunde bei einem Laden, in dem man Saft kaufen kann, um sich zu erfrischen. Jeder der vier Freunde hat bestimmte Vorlieben, die in der Tabelle unten festgehalten sind. Mehr Herzen bedeutet, dass das Getränk lieber getrunken wird. Beispielsweise mag Anna das Getränk  mit drei Herzen, das Getränk  aber nur mit einem Herzen. Daniel hingegen mag das Getränk  mit vier Herzen und das Getränk  nur mit einem Herz.

				
Anna				
Beat				
Christine				
Daniel				

Der Saftladen ist sehr beliebt, daher hat er von jedem der vier Getränke *nur noch je ein Glas* übrig. Wähle die Getränke für die vier Freunde so, dass die Anzahl der Herzen insgesamt möglichst gross ist.



Lösung

Die höchst erreichbare Anzahl Herzen ist 14, beispielsweise bei der folgenden Lösung:

Anna				
Beat				
Christine				
Daniel				

Um auf diese Lösung zu kommen, geht man idealerweise von Daniel aus. Er mag das Getränk mit vier Herzen, das alle anderen nur mit einem Herz mögen. Wenn man dann Beat oder Christine das Getränk gibt, können die beiden übrig gebliebenen (Anna und Christine respektive Anna und Beat) jeweils ihr zweitliebstes Getränk wählen.

Drei der vier Freunde mögen am liebsten . Da aber nur eines dieser Getränke vorhanden ist, müssen zwei sich mit ihrem Zweitwunsch zufrieden stellen. Es kann also keine Kombination mit mehr als $3 + 3 + 4 + 4 = 14$ Herzen geben.

Es gibt auch nur diese beiden Lösungen mit 14 Herzen, da alle anderen Lösungen von mindestens einem der Freunde verlangen, dass er sein drittliebstes Getränk mit 2 Herzen wählt, so dass maximal $2 + 3 + 4 + 4 = 13$ Herzen erreicht werden können.

Dies ist Informatik!

In dieser Aufgabe geht es darum, die Anzahl der Herzen (und damit die Zufriedenheit der vier Freunde) zu optimieren. *Optimierung* ist ein wichtiger Forschungsschwerpunkt der Informatik und auch der Mathematik, da sie an vielen Stellen auftaucht und häufig Algorithmen zum Finden optimaler Lösungen für eine Aufgabe sehr viel Zeit brauchen. In diesem Fall muss ein einfacher Algorithmus, der alle möglichen (und unmöglichen) Lösungen durchsucht, über 65000 verschiedene Lösungen ausprobieren. Durch geschicktes Überlegen kann man das zwar drastisch reduzieren (es gibt nur 24 mögliche Lösungen, für die die Anzahl der Herzen ausgerechnet werden muss), diese Überlegungen sind jedoch nicht immer offensichtlich.

Das konkrete Problem dieser Aufgabe ist eine Sonderform des *Matching-Problems*: jeder der vier Personen soll genau ein Getränk zugewiesen werden und es gibt von jedem der vier Getränke nur genau eines. Zudem soll die Zufriedenheit möglichst hoch sein. Solche Probleme tauchen in der Welt ebenfalls auf, man denke nur an die Wartelisten für Organtransplantationen. Auch hier muss den



Patienten ein Organ zugewiesen werden und gleichzeitig müssen *einschränkende Bedingungen* (wie beispielsweise die Wartezeit oder die Dringlichkeit der Transplantation aber auch die Verträglichkeit) in Betracht gezogen werden.

Webseiten und Stichwörter

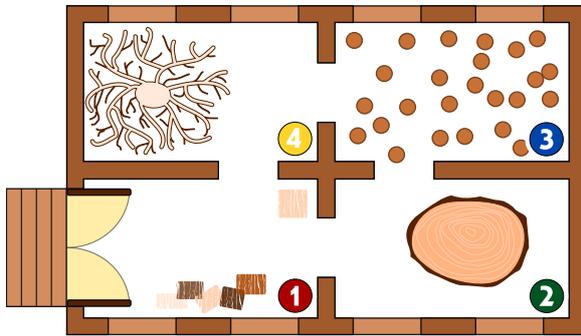
Optimierung, Matching

- [https://de.wikipedia.org/wiki/Optimierung_\(Mathematik\)](https://de.wikipedia.org/wiki/Optimierung_(Mathematik))
- <https://de.wikipedia.org/wiki/Branch-and-Bound>
- [https://de.wikipedia.org/wiki/Matching_\(Graphentheorie\)](https://de.wikipedia.org/wiki/Matching_(Graphentheorie))





24. Alarm im Museum!



Im Museum für moderne Holzplastiken gibt es ein intelligentes Sicherheitssystem. Da die modernen Holzplastiken interessant aussehen, bewegen sich die Personen sehr langsam im Museum.

Das System ermittelt in jeder Minute, wie viele Personen in jedem Raum sind, und trägt die Zahlen in eine Tabelle ein. Ausserdem prüft es anhand der Zahlen, ob ein Eindringling im Museum ist. Ein Eindringling ist eine Person, die in das Museum gelangt ist, aber nicht die Eingangstür benutzt hat.

Sobald das System meint, dass ein Eindringling im Museum ist, schlägt es Alarm.

Unten siehst Du die Tabelle des Systems für die Minuten 10:01 bis 10:07. Oben siehst du den Raumplan des Museums mit den Nummern der Räume.

Zeit	Raum 1	Raum 2	Raum 3	Raum 4
10:01	2	0	0	0
10:02	3	0	0	0
10:03	2	1	0	0
10:04	4	1	1	0
10:05	2	2	3	0
10:06	5	2	2	1
10:07	4	1	2	2

In welcher Minute in der Tabelle schlägt das System Alarm?

- A) 10:01
- B) 10:02
- C) 10:03
- D) 10:04
- E) 10:05
- F) 10:06
- G) 10:07



Lösung

Das System schlägt um 10:05 Alarm. Zu diesem Zeitpunkt waren in Raum 3 zwei Personen mehr als in der Minute davor. In den Nachbarräumen war da (also um 10:04) nur eine Person, nämlich in Raum 2. Folglich muss jemand von aussen in Raum 3 eingedrungen sein, ohne den Eingang des Museums zu benutzen.

Zu den anderen Zeitpunkten wird kein Eindringling erkannt:

- Um 10:01 sind 2 Personen in Raum 1; sie können den Eingang des Museums benutzt haben.
- Von 10:01 bis 10:02 kann 1 Person durch den Eingang des Museums gekommen sein.
- Von 10:02 bis 10:03 kann 1 Person von Raum 1 nach Raum 2 gegangen sein.
- Von 10:03 bis 10:04 können 1 Person von Raum 2 nach Raum 3 und eine Person von Raum 1 nach Raum 2 gegangen und 3 Personen durch den Eingang gekommen sein.
- Von 10:05 bis 10:06 können 1 Person von Raum 3 nach Raum 4 gegangen und 3 Personen durch den Eingang gekommen sein.
- Von 10:06 bis 10:07 können 1 Person von Raum 1 nach Raum 4 und 1 Person von Raum 2 nach Raum 1 gegangen sein sowie 1 Person das Museum durch den Eingang verlassen haben.

Dies ist Informatik!

Sicherheitssysteme, die die Anzahl von Personen in kritischen Bereichen überwachen, findet man zum Beispiel auf Flughäfen. Computerprogramme werten die Live-Bilder von Überwachungskameras aus, erkennen Personen, zählen sie und werten die Zahlen aus. Diese Programme setzen künstliche Intelligenz ein, zum Beispiel zum Erkennen menschlicher Wesen. Aber sie verwenden auch einfache Regeln wie in dieser Aufgabe, um Sicherheitsprobleme aufzuspüren.

Webseiten und Stichwörter

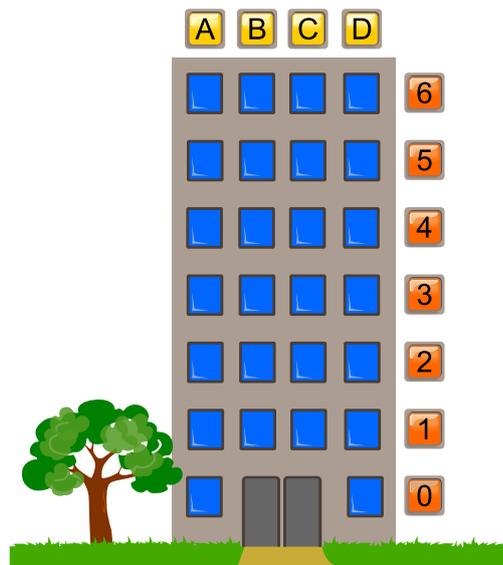
Intelligentes Sicherheitssystem, Regel, Zustandsübergang

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

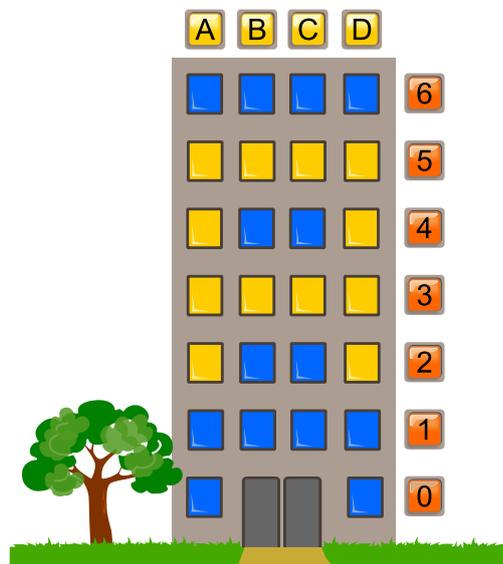


25. Lichtkunst

Das neue Hochhaus in der Stadt hat eine zentrale Anlage zum Ein- und Ausschalten der Lichter. Das Hochhaus hat 26 Fenster, hinter denen das Licht ein- und ausgeschaltet werden kann. Leider kann man aber nicht das Licht jedes Fensters einzeln ein- und ausschalten, sondern nur immer ein ganzes Stockwerk oder eine Fensterspalte.



Welche Stockwerknamen oder Spaltennamen musst Du schalten, so dass das Hochhaus am Ende so aussieht:





Lösung

Die Aufgabe kann man am einfachsten lösen, indem man zunächst die Stockwerke 3 und 5 anschaltet, danach die Fensterspalten A und D anschaltet und zuletzt die Stockwerke 6, 1 und 0 wieder ausschaltet. Natürlich gibt es noch viele andere Lösungen, die im Kern aber auf dieselbe Abfolge zurückgehen.

Dies ist Informatik!

Die Lichtschalter in dieser Aufgabe können mit Befehlen für ein beliebiges System oder eine beliebige Maschine verglichen werden. Man kann sogar so weit gehen und das Ausführen von komplizierten Computerprogrammen als solch einfache Befehle verstehen. Auch wenn es noch so kompliziert erscheint, so werden am Ende doch sehr wenige, einfache Befehle ausgeführt. Die Fenster entsprechen Speicherplätzen, in denen entweder 0 („aus“) oder 1 („an“) gespeichert werden kann.

Moderne Computer erlauben es, zusammengesetzte Befehle auszuführen, die mehrere Variablen gleichzeitig bearbeiten. Das sind in diesem Fall die Lichtschalter für ein ganzes Stockwerk oder eine Fensterspalte. Moderne Computer arbeiten in der Regel mit Millionen von Speicherplätzen gleichzeitig, und zwar alleine im Prozessor . . . der Arbeitsspeicher oder die Festplatte sind da noch viel grösser, dort sind Milliarden oder Billionen von einzelnen Speicherplätzen vorhanden.

Daher ist es wichtig, dass für die zusammengesetzten Befehle klar definiert ist, wann sie ausgeführt werden können (*pre-condition*), und was nach ihrem Ausführen gelten soll (*post-condition*). Im Beispiel des Änderns der Beleuchtung eines Stockwerks gilt: wenn nur ein Licht ausgeschaltet ist (*pre-condition*), sind nachher alle Lichter des Stockwerks angeschaltet (*post-condition*). Ansonsten (also: alle Lichter sind eingeschaltet, *pre-condition*) werden alle Lichter des Stockwerks ausgeschaltet (*post-condition*).

Webseiten und Stichwörter

Hardwarenahes Programmieren, Sequenzen, Binäroperationen

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



26. Ersetzungen

Herr Müller ist plötzlich erkrankt. In seiner Firma soll Herr Maier deshalb alle Aufgaben von Herrn Müller übernehmen. Zum Glück wurde Herr Müller nach 2 Wochen wieder gesund, weil aber Herr Maier schon sehr gut eingearbeitet war, vereinbarten beide, dass Herr Müller jetzt die früheren Aufgaben von Herrn Maier übernehmen und Herr Maier die Aufgaben von Herrn Müller weiterführen soll. Die Projektdokumentation soll nun so geändert werden, dass im Text der Name Müller durch den Namen Maier ersetzt wird und umgekehrt. Im verwendeten Projektplaner können beliebige Texte durch andere ersetzt werden.

Welche Vorgangsweise ist sinnvoll, wenn man annimmt, dass im Text nirgends ein „#“-Zeichen vorkommt?

- A) Ich ersetze zuerst alle „Müller“ mit „Maier“ und dann alle „Maier“ mit „Müller“.
- B) Ich ersetze zuerst alle „Maier“ mit „Müller“ und dann alle „Müller“ mit „Maier“.
- C) Ich ersetze alle „Müller“ mit „#“ dann alle „#“ mit „Maier“ und dann alle „Maier“ mit „Müller“.
- D) Ich ersetze alle „Müller“ mit „#“ dann alle „Maier“ mit „Müller“ und dann alle „#“ mit „Maier“.



Lösung

Die richtige Antwort ist D) Ich ersetze alle „Müller“ mit „#“ dann alle „Maier“ mit „Müller“ und dann alle „#“ mit „Maier“.

- A) In diesem Fall würden nur die „Müller“ überbleiben und alle „Maier“ verloren gehen, denn nach der ersten Ersetzung kommt nur noch „Maier“ vor, was dann komplett durch „Müller“ ersetzt würde.
- B) In diesem Fall würden nur die „Maier“ überbleiben und alle „Müller“ verlorengehen, denn nach der ersten Ersetzung kommt nur noch „Müller“ vor, was dann komplett durch „Maier“ ersetzt würde.
- C) In diesem Fall würden nur die „Müller“ überbleiben und alle „Maier“ verlorengehen, denn nachdem alle „Müller“ durch „#“ ersetzt wurden, werden diese sofort weiter durch „Maier“ ersetzt, welche wiederum komplett durch „Müller“ ersetzt werden.
- D) Ist die einzige Vorgangsweise die funktioniert, weil die Vorkommen von Müller durch „#“ ersetzt werden und so erhalten bleiben, während die „Maier“ durch „Müller“ ersetzt werden.

Dies ist Informatik!

Obwohl eine einzelne Ersetzung ein ganz einfacher Vorgang ist, spielen Ersetzungen eine wichtige Rolle in der Informatik. Mittels Serien von Ersetzungen können komplexe Aufgaben durchgeführt werden. Auch in der theoretischen Informatik kommen sie vor, so werden zum Beispiel formale Grammatiken als eine Liste von Ersetzungsregeln formuliert.

In dieser Aufgabe besteht die Schwierigkeit darin, dass zwei Begriffe miteinander vertauscht werden ... dies funktioniert nur, wenn ein dritter sonst nicht verwendeter Begriff als temporärer Begriff verwendet wird.

Webseiten und Stichwörter

Textverarbeitung, Sequenzen von Anweisungen befolgen, Variablentausch

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



27. Wege durch den Irrgarten

Benj möchte durch einen Irrgarten gehen. Er bittet Dich, ihm zu sagen, wie er durch den Irrgarten gehen kann. Er betritt den Irrgarten beim schwarzen Dreieck und möchte den Ausgang beim roten Kreis erreichen. Benj kann sich aber nur acht der folgenden Schritte merken:

		Gehe einen Schritt geradeaus und drehe Dich dann nach links.
		Gehe einen Schritt geradeaus und drehe Dich dann nach rechts.
		Gehe einen Schritt geradeaus.

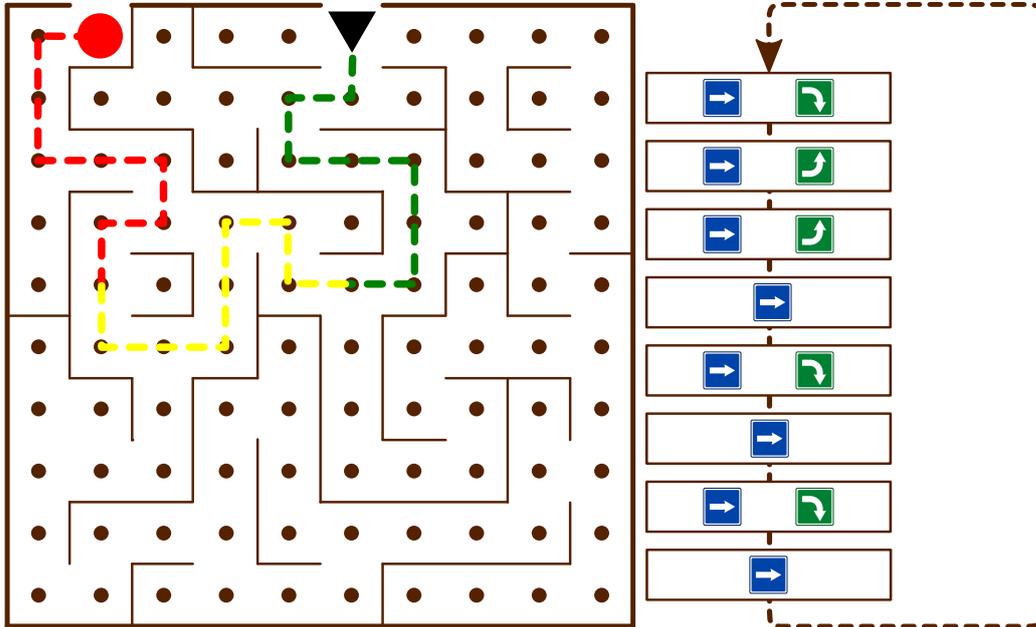
Auch wenn sich Benj nur acht Schritte merken kann, kann er diese acht Schritte wiederholt durchführen.

Am Anfang schaut Benj wie das schwarze Dreieck nach unten. Wähle die Schritte in der richtigen Reihenfolge für die leeren Felder, so dass Benj den Ausgang beim roten Punkt findet.



Lösung

Die folgende Abfolge von Befehlen führt zum Ausgang, wenn sie dreimal ausgeführt wird:



Dies ist Informatik!

Benj führt ein Programm aus. Dieses Programm besteht aus einer Abfolge von Befehlen („*Sequenz*“). Eine *Kontrollstruktur* wie die *Schleife* in diesem Programm erlaubt, dass eine Abfolge von Befehlen mehrfach hintereinander ausgeführt wird, so oft wie benötigt. So muss man nicht dieselbe Abfolge von Befehlen häufig hintereinander kopieren sondern spart sich Arbeit. Auch können Fehler im Programm so einfacher gefunden und korrigiert werden.

Webseiten und Stichwörter

Sequenz, Schleife, Algorithmus

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

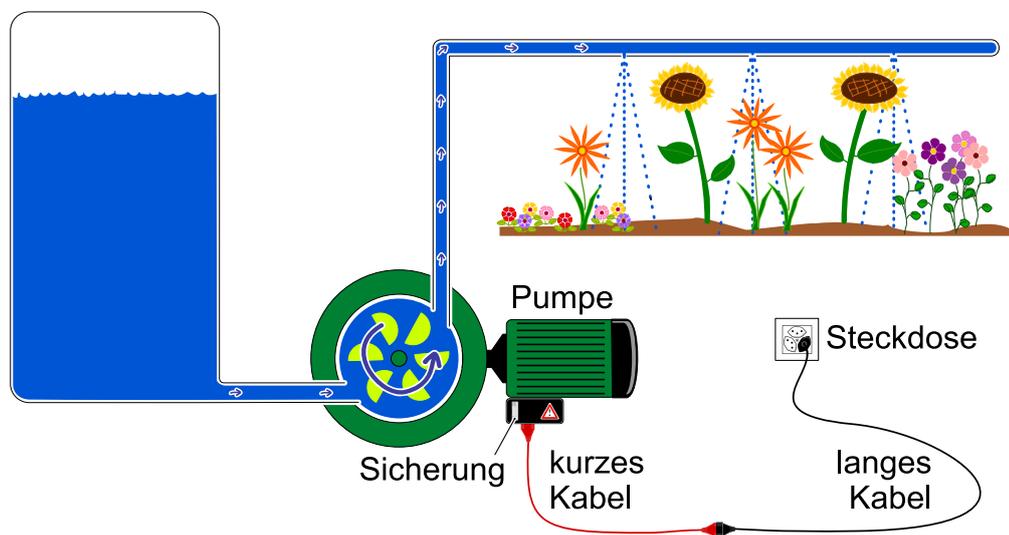


28. Bewässerungssystem

Gustav hat ein Blumenbeet und einen Gemüsegarten. Dafür hat er zwei gleiche Bewässerungssysteme gebaut. Eines davon kann man unten sehen.

Es ist mit dem Stromnetz verbunden und besteht aus:

- einem langen Kabel
- einem kurzen Kabel
- einer Pumpe
- die Pumpe hat eine Sicherung (die Pumpe funktioniert nicht, wenn die Sicherung raus geflogen ist)



Eines Tages funktioniert das Bewässerungssystem des Blumenbeets nicht mehr. Gustav stellt fest, dass es nicht an dem Wassertank und an den Wasserleitungen liegt.

Lediglich ein Teil ist defekt. Gustav möchte den Fehler finden. Er kann dazu alle Teile des zweiten Bewässerungssystems (dem aus dem Gemüsegarten) verwenden. Durch den Austausch von Teilen will er den Fehler finden.

Kreuze alle Aussagen an, die korrekt sind.

- Wenn er etwas testet, muss er mit der Pumpe beginnen, denn sie ist der wichtigste Teil.
- Schrittweise können beliebig viele Teile vom zweiten Bewässerungssystem verwendet werden, aber in jedem Schritt sollte nur ein Teil ausgetauscht werden. Wenn das System dann wieder funktioniert, dann war das letzte ausgetauschte Teil fehlerhaft.
- Zuallererst kann man mit einem anderem Gerät testen, ob die Steckdose vielleicht stromlos ist. Wenn diese funktioniert, kann man sich schrittweise vorarbeiten.
- Man sollte lieber neue Teile besorgen. Die vom Gemüsegarten sind ja schon gebraucht.
- In jedem Schritt sollte man mindestens zwei Teile austauschen. So kommt man schneller zum Ziel.



Lösung

Die Antworten B) und C) sind richtig.

Zur Antwort A): wie wichtig ein Teil ist, sagt nichts aus über die Fehlerwahrscheinlichkeit

Zur Antwort D): auch ganz neue Teile sind manchmal defekt. Die Teile aus dem Gemüsegarten funktionieren aber ganz sicher, weil die zweite Bewässerung aktuell ja funktioniert.

Zur Antwort E): das wäre wirklich schneller, wenn es nur darum ginge das System schnell zu reparieren. Er könnte ja sogar einfach die ganze Bewässerungsanlage vom Gemüsegarten übernehmen. Gustav möchte aber das fehlerhafte Teil finden. Wenn er mehrere Teile gleichzeitig austauscht, muss er noch weitere Tests machen, um herauszufinden, welches Teil nun defekt ist.

Dies ist Informatik!

Fehlerbehebung, auch *Debugging* genannt, ist eine wichtige Aufgabe, weil bei komplexen Anlagen früher oder später Fehler auftreten können. Einen Fehler zu finden, ist schon eine Herausforderung, besonders schwierig ist es aber zwei oder mehr Fehler gleichzeitig zu finden. Um das zu vermeiden haben Entwickler die Strategie nach jedem Änderungsschritt, wenn möglich, Tests durchzuführen. Dadurch erhöht man die Chance, dass man nur nach einem einzelnen Fehler suchen muss.

Webseiten und Stichwörter

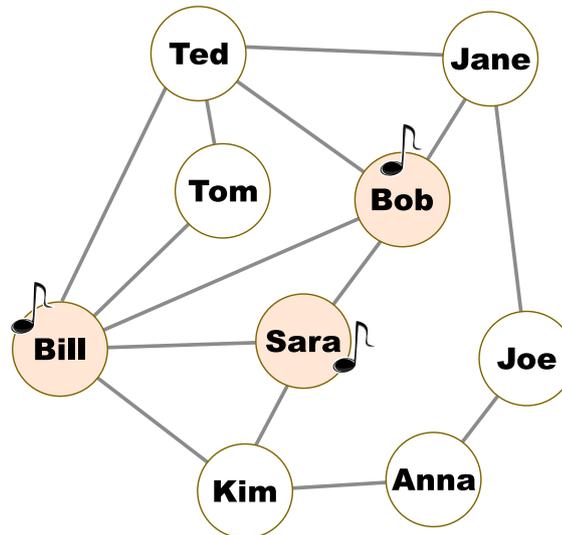
Fehlersuche, Debugging

- <https://en.wikipedia.org/wiki/Systemantics>
- https://en.wikiquote.org/wiki/John_Gall
- <https://en.wikipedia.org/wiki/Debugging>



29. Der neue Song

Im folgenden Diagramm sind zwei Personen Freunde, wenn ihre Namen durch eine Linie verbunden sind; sonst nicht. Am Montag veröffentlicht ein Megastar seinen neuen Song. Noch am selben Tag kaufen Bill, Bob und Sara den neuen Song. Ihre Namen sind mit einer Note markiert.



Ab Dienstag passiert jeden Tag Folgendes: Genau die Personen kaufen den Song, für die gilt, dass mindestens die Hälfte ihrer Freunde den Song schon am Vortag hatte. Am Dienstag kauft z.B. Tom den Song, Jane aber nicht.

An welchem Tag haben alle in der Gruppe frühestens den neuen Song?

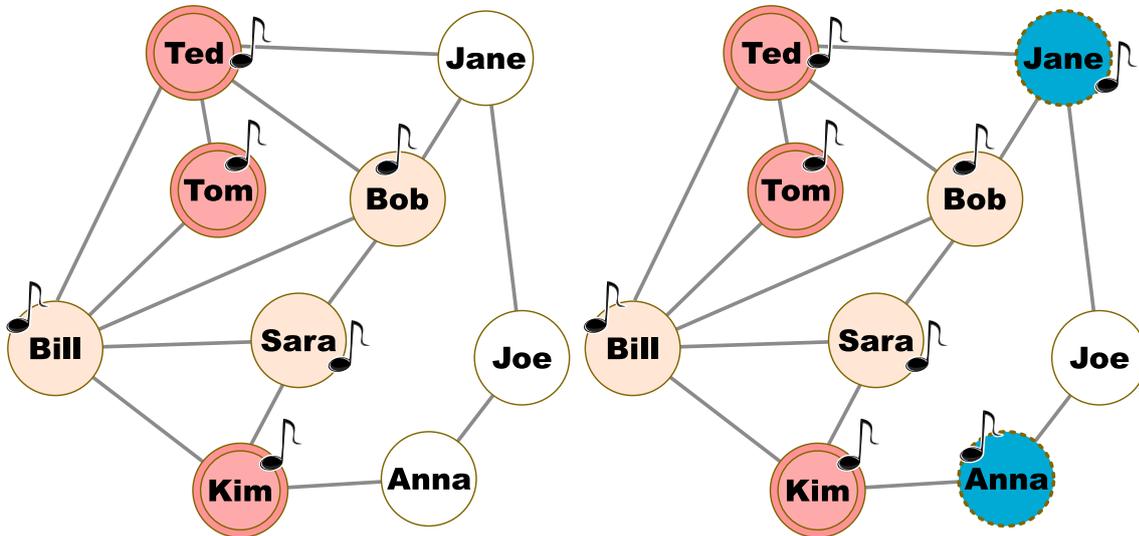
- A) Mittwoch
- B) Donnerstag
- C) Freitag
- D) Samstag



Lösung

Antwort B) ist richtig.

Tom, Ted und Kim kaufen den Song am Dienstag, Anna und Jane kaufen ihn am Mittwoch:



Joe kauft daraufhin den Song am Donnerstag. An diesem Tag haben alle in der Gruppe den Song; vorher nicht.

Dies ist Informatik!

So genannte „Soziale Netzwerke“ haben manchmal Milliarden Mitglieder. Deshalb sind sie nicht nur für ihre Benutzerinnen und Benutzer interessant, sondern vor allem auch für Anbieter von Produkten. In Marketingkampagnen wird der Diffusionsprozess in sozialen Netzwerken ausgenutzt, um neue Produkte bekannt zu machen. Um diesen Prozess geht es in dieser Aufgabe.

Hier wird ein Schwellenmodell verwendet, um die Diffusion zu simulieren. Das Diagramm in der Aufgabe ist ein Graph. Er besteht aus Knoten (Namen von Personen) und Kanten (Linien zwischen den Knoten, die Freundschaften – im Sinne des sozialen Netzwerkes – zwischen Personen darstellen). Knoten, die über eine Kante direkt miteinander verbunden sind, nennt man Nachbarn. Die Infektionsschwelle q ist eine Zahl zwischen 0 und 1. In der Aufgabe ist $q = 0,5$. Eine Person zeigt ein neues Verhalten B (z.B. Kauf eines Produktes), wenn mindestens der Anteil q ihrer Nachbarn im Graph auch dieses Verhalten B zeigt.

Webseiten und Stichwörter

Soziales Netzwerk, Graph

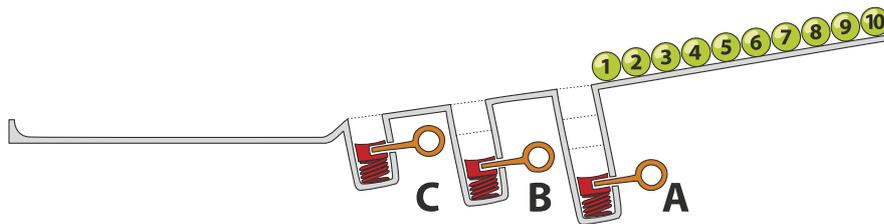
- <https://de.wikipedia.org/wiki/Kleine-Welt-Phänomen>



30. Kugelbahn

Auf einer Rampe liegen 10 nummerierte Kugeln. In der Rampe sind drei Löcher A, B und C, wobei A Platz für drei Kugeln hat, B Platz für zwei Kugeln und C Platz für eine Kugel. Wenn die Kugeln die Rampe hinunterrollen, füllen sie zuerst die Löcher (Kugeln 1, 2 und 3 in Platz A, Kugeln 4 und 5 in Platz B und Kugel 6 in Platz C). Der Rest rollt weiter.

Danach werden die Federn in den Löchern gelöst, zuerst von Platz A, dann von Platz B und zuletzt von Platz C. Dabei werden die Kugeln wieder auf die Rampe zurückgeschoben. Bevor eine Feder gelöst wird, wird gewartet bis alle anderen Kugeln vorbeigerollt sind.



In welcher Reihenfolge liegen die Kugeln am Ende?

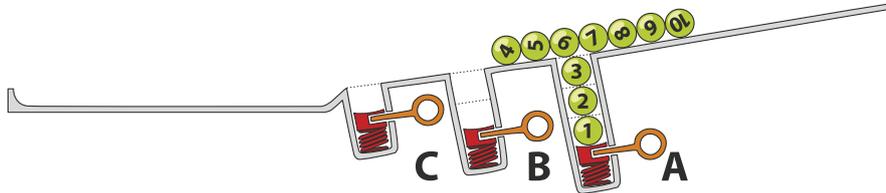
- A) B) C) D)



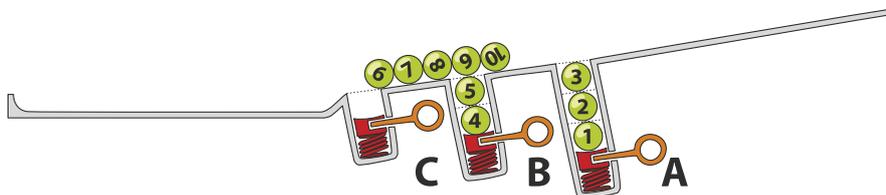
Lösung

Die richtige Antwort ist D) 7 8 9 10 3 2 1 5 4 6.

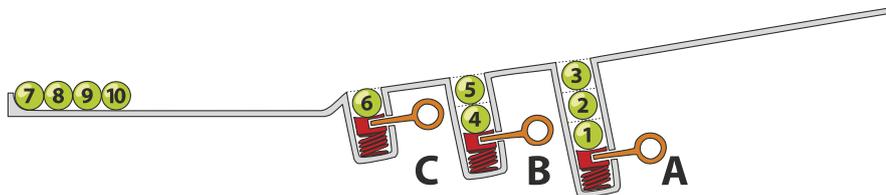
Die Kugeln 1, 2 und 3 fallen in Loch A, die Kugeln 4 bis 10 rollen über sie hinweg.



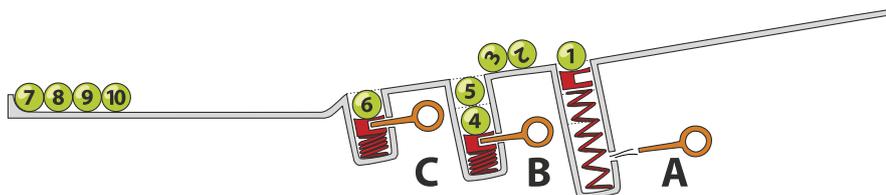
Kugeln 4 und 5 fallen in Loch B und Kugeln 6 bis 10 rollen über sie hinweg.



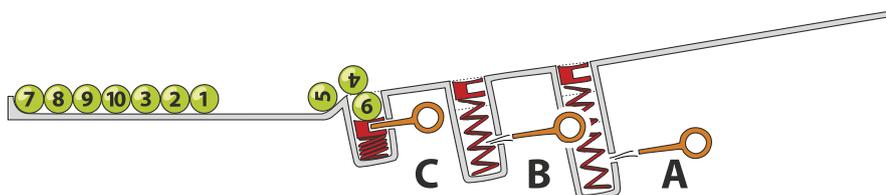
Schliesslich fällt Kugel 6 in Loch C und die Kugeln 7 bis 10 rollen in dieser Reihenfolge bis an das Ende der Rampe.



Nach Lösen der Feder in Loch A werden die Kugeln in der Reihenfolge 3, 2, 1 hinausgeschoben und rollen weiter bis an das Ende der Rampe.

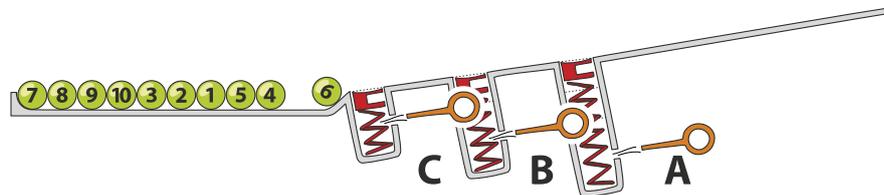


Nun sind die Kugeln in der Reihenfolge 7, 8, 9, 10, 3, 2, 1 am Ende der Rampe. Dann wird die Feder in Loch B gelöst, so dass 5 und 4 an das Ende rollen und zuletzt wird Kugel 6 aus Loch C hinausgeschoben.





Dadurch sind die Kugeln am Ende der Rampe in der Reihenfolge 7, 8, 9, 10, 3, 2, 1, 5, 4, 6 angekommen.



Dies ist Informatik!

Die Löcher in dieser Aufgabe ähneln der Datenstruktur Stapel (engl. *stack*). Ein Stapel ist eine Möglichkeit Daten zu speichern, die auf dem *Last-In First-Out Prinzip (LIFO)* basiert. Die letzte Kugel, die hineinfällt, wird zuerst wieder ausgeworfen. Auch wenn dieses Prinzip sehr einfach erscheint, ist es doch in vielen Situationen sehr nützlich. Will man beispielsweise automatisch feststellen, ob die Klammern in einem arithmetischen Ausdruck richtig gesetzt sind (in $((1 + 2) \cdot 3)$ sind sie richtig, in $((4 + 5) \cdot (6 - 7))$ sind sie falsch), geht man von links nach rechts so vor: öffnende Klammern auf den Stack geben (mit einer sogenannten „*push*“-Operation), wenn eine passende schliessende Klammer gefunden wird, die öffnende Klammer wieder vom Stack entfernen (mit einer sogenannten „*pop*“-Operation). Wenn eine nicht passende schliessende Klammer gefunden wird, ist ein Fehler im Ausdruck, wenn am Ende der Stapel leer ist, gab es zu jeder öffnenden Klammer auch eine passende schliessende Klammer.

Webseiten und Stichwörter

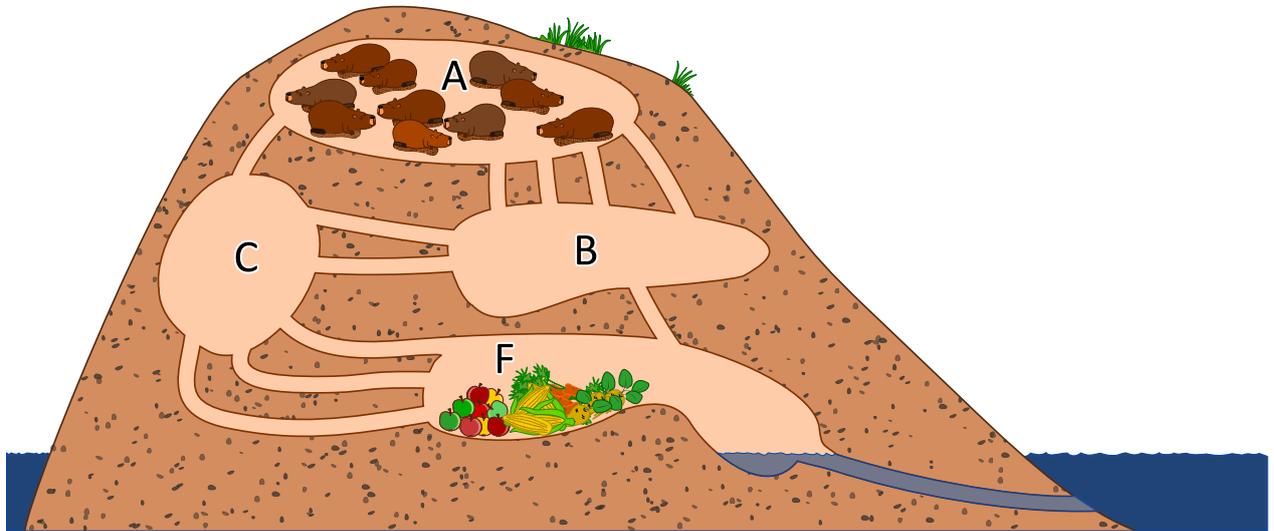
Stack (dt. Stapel), LIFO

- https://de.wikipedia.org/wiki/Last_In_-_First_Out
- <https://de.wikipedia.org/wiki/Stapelspeicher>





31. Gänge im Biberbau



10 Biber befinden sich in Raum A und wollen möglichst rasch in den Raum F zum Fressen kommen. Ein Biber braucht 1 Minute um durch einen Gang zu laufen. Leider kann durch jeden Gang immer nur ein Biber gleichzeitig laufen. Sie können dabei also nicht direkt hintereinander durch den Gang laufen. In den Räumen A, B, C und F ist genug Platz für alle Biber und das Durchqueren eines Raums benötigt keine Zeit.

Nach wieviel Minuten können alle 10 Biber in Raum F sein? Gib die kürzest mögliche Zeit an!



Lösung

Alle 10 Biber können bereits nach 4 Minuten in Raum F sein.

Im Biberbau gibt es zwei kürzeste Wege zum Raum F. Beide Wege können innerhalb von 2 Minuten nur jeweils einen Biber ans Ziel bringen, in 3 Minuten sind es jeweils 2 Biber:

- $A \rightarrow B \rightarrow F$
- $A \rightarrow C \rightarrow F$

Der Weg $A \rightarrow B \rightarrow C \rightarrow F$ bietet Platz für 2 Biber, dauert aber 3 Minuten. Nach 3 Minuten haben wir dann insgesamt erst 6 Biber in Raum F. Wir brauchen also noch eine 4. Minute. Wie z.B. alle 10 Biber in 4 Minuten ans Ziel kommen können, zeigt folgende Tabelle:

Aktion / Situation	Anzahl Biber in den Räumen (nach der Aktion)			
	A	B	C	F
Situation zu Beginn	10	0	0	0
<i>3 Biber gehen von A nach B (weniger als möglich ist)</i> <i>1 Biber geht von A nach C</i>				
Situation nach 1 Minute	6	3	1	0
<i>3 Biber gehen von A nach B (weniger als möglich ist)</i> <i>1 Biber geht von B nach F</i> <i>2 Biber gehen von B nach C</i> <i>1 Biber geht von C nach F</i> <i>1 Biber geht von A nach C</i>				
Situation nach 2 Minuten	2	3	3	2
<i>1 Biber geht von A nach B (kürzester Weg)</i> <i>1 Biber geht von B nach F</i> <i>2 Biber gehen von B nach C</i> <i>1 Biber geht von A nach C (kürzester Weg)</i> <i>3 Biber gehen von C nach F</i>				
Situation nach 3 Minuten	0	1	3	6
<i>1 Biber geht von B nach F</i> <i>3 Biber gehen von C nach F</i>				
Situation nach 4 Minuten	0	0	0	10

Es gibt verschiedene Lösungswege, wie die Biber innerhalb von 4 Minuten in Raum F kommen können. Bei der hier gezeigten Lösung müssen die Biber in keinem Raum auf das Weitergehen warten.

Dies ist Informatik!

Das Netzwerk der Gänge kann als sogenanntes Fluss-Netzwerk aufgefasst werden. Die Anzahl der Gänge zwischen zwei Räumen bestimmt wie viele Biber innerhalb einer Minute von einem zum anderen Raum gehen können. Es ist die sogenannte Kapazität der Verbindung zwischen zwei Räumen, der den Fluss zwischen diesen Räumen begrenzt.

Ein Fluss-Netzwerk ist in der Graphentheorie ein gerichteter Graph, bei dem jede Kante eine Kapazität hat (die Anzahl der Gänge in unserer Aufgabe). Ein Fluss, der durch die Kanten des Graphen fließt, ist durch die Kapazität der Kanten begrenzt. Mit Hilfe von Flüssen in Netzwerken kann ein



Computernetzwerk oder ein Verkehrsnetzwerk simuliert werden und die Engpässe in den Verkehrsströmen aufgedeckt werden.

In Fluss-Netzwerken ist insbesondere der maximal mögliche Fluss zwischen zwei Knoten interessant. In unserer Aufgabe wären das 4 Biber, die pro Minute von Raum A zum Raum F ohne Wartezeit in einem der Räume laufen können. Der sogenannte Ford-Fulkerson Algorithmus kann so einen maximalen Fluss berechnen.

Webseiten und Stichwörter

Graph, Fluss in Netzwerken

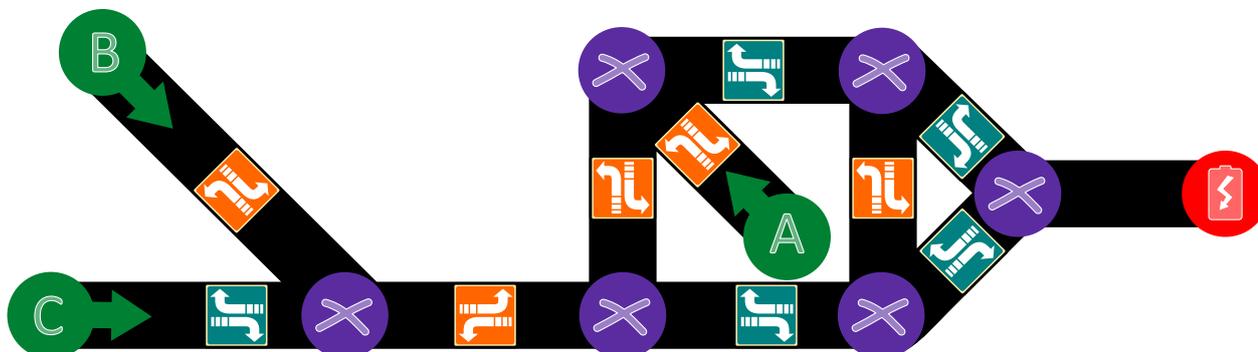
- https://de.wikipedia.org/wiki/Flüsse_und_Schnitte_in_Netzwerken
- https://de.wikipedia.org/wiki/Gerichteter_Graph
- https://de.wikipedia.org/wiki/Algorithmus_von_Ford_und_Fulkerson





Lösung

Die richtige Lösung ist:



Für die beiden Linien ganz links im Bild von B und C muss sichergestellt werden, dass der Arabot weiter nach rechts im Bild fahren kann, also muss er oben und unten wählen.

Im rechten Teil des Bildes ist die einzige Möglichkeit, zur Ladestation () zu fahren, wenn er die untere Linie von links her durchfährt. Im Quadrat in der Mitte des Bildes muss also für die rechte Linie gewählt werden, damit von oben her der Einstieg funktioniert. Ebenso muss man für die untere Linie im mittleren Teil des Bildes ein wählen.

Falls der Arabot diese beiden Linien in entgegengesetzter Richtung durchfährt, muss verhindert werden, dass er bei A landet, also muss im mittleren Teil des Bildes oben ein und links ein gewählt werden. Der Aufbau der Linien hat zur Folge, dass der Arabot dann letztlich wieder auf den rechten Teil des Bildes zurückfährt, so dass er auch in diesen Fällen an der Ladestation () landet.

Dies ist Informatik!

In dieser Aufgabe geht es darum, verschiedene mögliche Wege zu einem Ziel (A zu , B zu und C zu) in einer einzelnen beschrifteten Struktur (ein Graph in diesem Fall) zu kodieren. In der Informatik nennt man dies eine *Datenstruktur*. Wenn man einen Weg entlang geht (beispielsweise von A zu), muss der Arabot Schritt für Schritt die Anweisungen lesen und ausführen: „In welche Richtung drehe ich mich an der nächsten Kreuzung? Wenn ich das weiss, gehe ich diesen Weg.“ Ein Computer arbeitet auf der Ebene der Hardware ähnlich: Befehl lesen, ausführen und so weiter.

Hinter dieser Aufgabe stehen viele interessante mathematische und informatische Fragen, die damit zu tun haben, wie schwierig es ist, solche Beschriftungen korrekt und eindeutig festzulegen. Einige solcher Fragen sind bisher ungelöst und werden momentan von Informatikern im Fachgebiet der *Algorithmen* und der *Komplexitätstheorie* erforscht. Vergleichbare Aufgabenstellungen gibt es in den Bereichen der *Computational Biology* sowie der *Computational Medicine*.

Webseiten und Stichwörter

Bidirektionale Graphen, Komplexitätstheorie, Computational Biology, Computational Medicine

- <https://de.wikipedia.org/wiki/Komplexitätstheorie>
- https://en.wikipedia.org/wiki/Computational_biology
- https://en.wikipedia.org/wiki/In_silico_medicine



33. Zahnstocherteilete

Helga und Bob spielen ein Spiel mit Zahnstochern. Am Anfang liegen zwei Haufen Zahnstocher auf dem Tisch. Immer wenn einer der beiden an der Reihe ist...

1. ...legt der Spieler einen der beiden Haufen beiseite...
2. ...und teilt den restlichen Haufen in zwei Teile.

Wenn ein Spieler am Ende zwei Haufen mit jeweils einem Zahnstocher übrig lässt, hat er gewonnen. Helga fängt an.

Am Anfang wählt Helga den Haufen mit 24 Zahnstochern, den sie nun in zwei Stapel teilen muss. Wähle alle Teilungen aus, die Helga einen Sieg ermöglichen:

- A) 11 und 13
- B) 12 und 12
- C) 7 und 17
- D) 8 und 16



Lösung

Helga muss entweder 11 und 13 oder 7 und 17 wählen, damit sie gewinnen kann.

Damit Helga das Spiel gewinnen kann, ist es für sie eine gute Strategie, zwei Haufen mit einer ungeraden Anzahl von Zahnstochern zu hinterlassen. Wenn sie einen Haufen mit einer geraden Anzahl von Zahnstochern hinterlässt, kann Bob gewinnen. 24 Zahnstocher kann man natürlich nicht in einen Haufen mit einer geraden und einen mit einer ungeraden Anzahl von Zahnstochern aufteilen. Warum ist das so? Wenn ein Spieler zwei Haufen mit einer ungeraden Anzahl von Zahnstochern hat, kann er den von ihm gewählten Haufen nur einen Haufen mit einer geraden und einer ungeraden Anzahl von Zahnstochern einteilen. Der nächste Spieler entfernt dann den Haufen mit der ungeraden Anzahl von Zahnstochern und teilt den Haufen mit der geraden Anzahl von Zahnstochern wieder in zwei Haufen mit einer ungeraden Anzahl von Zahnstochern. Das Spiel ist beendet, wenn zwei Haufen mit je einem Zahnstocher übrig sind ... das ist eine ungerade Anzahl von Zahnstochern. Damit gewinnt der Spieler, der zwei Haufen mit einer ungeraden Anzahl von Zahnstochern hinterlassen hat.

Dies ist Informatik!

Für Spiele wie dieses können Strategien leicht gefunden werden, wenn man eine invariante (sich während des Spielverlaufs nicht verändernde) Eigenschaft findet, die in jedem Fall zum Sieg führt. Das ist in diesem Fall die Tatsache, dass nach dem Zug zwei Haufen mit einer ungeraden Anzahl von Zahnstochern hinterlassen werden.

In diesem Spiel ist es so, dass nicht nur eine geeignete Strategie erforderlich ist, man muss auch durch das Festlegen der Startposition verhindern, dass der Gegner die Strategie für sich ausnutzt. So legt die Startposition oder die Frage, wer anfängt, oftmals auch fest, wer das Spiel gewinnt, wenn beide Spieler optimal spielen.

Informatiker beschäftigen sich oft mit solchen Spielen, in denen kein Zufall vorkommt, sondern alleine die Strategie über Gewinn und Verlust entscheidet. Von kleinen Spielen wie diesem, das innerhalb von ein paar Sekunden von einem einfachen Computer berechnet werden kann bis hin zu grossen Spielen wie Schach oder Go, wo selbst die grössten Computer in vielen Jahren nicht den „besten“ Zug ausrechnen können, lernt man so, in komplexen Situationen die richtigen Entscheidungen zu treffen ... sei es am Ende nur in einem Computerspiel oder in Anwendungen künstlicher Intelligenz.

Webseiten und Stichwörter

Strategiespiel, Spielbaum

- https://en.wikipedia.org/wiki/Abstract_strategy_game
- https://de.wikipedia.org/wiki/Kombinatorische_Spieltheorie



34. Wortabstände

Um den Abstand zwischen zwei Worten herauszufinden, darf man die folgenden Schritte machen:

- einen Buchstaben an einer beliebigen Stelle hinzufügen
- einen Buchstaben an einer beliebigen Stelle entfernen
- einen Buchstaben an einer beliebigen Stelle durch einen anderen Buchstaben ersetzen

Der Abstand zwischen zwei Worten ist die Mindestanzahl von solchen Schritten, um von dem einen Wort zum anderen Wort zu kommen.

Der Abstand zwischen „rennen“ und „stehen“ ist damit 4, wie man am folgenden Beispiel sieht:

1. rennen → sennen („r“ durch „s“ ersetzen)
2. sennen → stennen („t“ einfügen)
3. stennen → stehen („n“ durch „h“ ersetzen)
4. stehen → stehen („n“ löschen)

Was ist der Abstand zwischen Emil und Erich?



Lösung

Der Abstand zwischen Emil und Erich ist 3, denn man kann zum Beispiel folgende Schritte machen:

Emil → Eril → Eric → Erich

Mit weniger als drei Schritten geht es nicht, denn Erich ist um einen Buchstabe länger (ein Schritt) und in Erich sind weder ein „m“ noch ein „l“ (zwei Schritte).

Dies ist Informatik!

Diese Wortabstände nennt man *Levenshtein-Distanz*, da der Russe Vladimir Levenshtein sie 1965 zuerst beschrieben hat. Sie wird beispielsweise verwendet, um in bei Rechtschreibhilfen korrekte Schreibweisen vorzuschlagen: wenn die Levenshtein-Distanz zwischen dem falsch geschriebenen Wort und der vorgeschlagenen richtigen Schreibweise klein ist, hat man wahrscheinlich nur einen Tippfehler gemacht. Ähnliche Abstände werden häufig verwendet, beispielsweise bei der Ähnlichkeit von DNA-Strängen, Bildern oder auch beim automatisierten Übersetzen von Texten.

Man kann die Levenshtein-Distanz mit dem Computer berechnen, indem man alle möglichen Ersetzungen ausprobiert. Damit der Computer dabei nicht unnötig lange Worte ausprobiert, baut man Grenzen ein.

Webseiten und Stichwörter

Levenshtein-Distanz, Editierdistanz

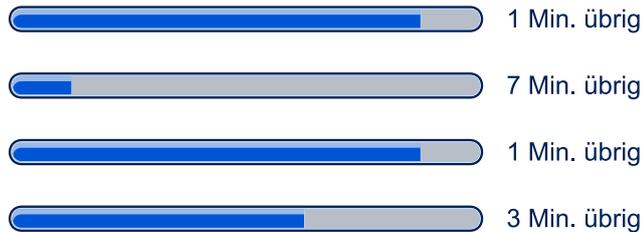
- <https://de.wikipedia.org/wiki/Levenshtein-Distanz>
- https://en.wikibooks.org/wiki/Algorithm_Implementation/Strings/Levenshtein_distance



35. Mehrere Downloads

Lädt man mehrere grössere Dateien herunter, dann teilen sich diese Downloads die Kapazität der Verbindung. Beim Herunterladen von 10 Dateien gleichzeitig kann jede Datei nur einen Zehntel der Verbindungskapazität nutzen.

Ein Benutzer lädt gerade 4 Dateien gleichzeitig herunter, die verbleibende Zeit wird lediglich aufgrund der aktuellen Geschwindigkeit berechnet:



Wie viele Minuten wird es dauern bis alle Dateien fertig heruntergeladen sind?



Lösung

Es wird 3 Minuten dauern.

Nach einer Minute sind 2 der Dateien fertig heruntergeladen. Bei den anderen beiden Dateien verbleiben 6 und 2 Minuten. Die Downloadgeschwindigkeit verdoppelt sich aber weil die verbleibenden Dateien jetzt die doppelte Kapazität nutzen können. Es verbleiben also 3 Minuten und 1 Minute:



Nach einer weiteren Minute ist die dritte Datei fertig heruntergeladen. Die letzte Datei hätte jetzt noch verbleibende 2 Minuten aber die Geschwindigkeit verdoppelt sich wieder und daher verbleibt genau 1 Minute:



Nach insgesamt 3 Minuten sind alle 4 Dateien heruntergeladen.

Dies ist Informatik!

Ein *Fortschrittsbalken* (engl. *progress bar*) ist ein Steuerelement, das anzeigt, wie weit die Bearbeitung eines Auftrags, wie zum Beispiel einem Installations- oder Ladevorgang, fortgeschritten ist. In der Regel wird zudem geschätzt, wie lange der Auftrag noch dauert. Ein Sonderfall ist eine Anzeige, bei der nicht bekannt ist, wie lange die Fertigstellung noch dauert, aber trotzdem deutlich gemacht werden soll, dass der Computer aktiv weiterarbeitet und nicht hängengeblieben ist. In diesem Fall wird entweder ein unbestimmter Fortschrittsbalken, eine Sanduhr oder ein rotierendes Symbol angezeigt.

Die Datenübertragungsrate, auch als Übertragungsgeschwindigkeit und nicht ganz zutreffend als „Bandbreite“ bezeichnet, ist die digitale Datenmenge, die innerhalb einer Zeitspanne über einen Übertragungskanal übertragen wird. Die maximal mögliche Datenübertragungsrate für einen Kanal wird als Kanalkapazität bezeichnet. Für einen Übertragungskanal ist aber auch die sogenannte Antwortverzögerung oder Latenzzeit ein wichtiges Mass. Damit wird die Zeitdauer angegeben, wie lange die Daten vom Sender zum Empfänger brauchen (manchmal wird auch der Rückweg zum Sender mit einberechnet).

Im Internet gibt es mehrere Faktoren, die die maximale Datenübertragungsrate zwischen zwei Computern bestimmen: Zunächst kann ein Server Daten nur mit einer maximalen Geschwindigkeit liefern, die von seiner Hardwareausstattung und Softwarekonfiguration abhängt. Dann ist der Server mit einer Leitung mit beschränkter Datenübertragungsrate mit dem Internet verbunden. Alle gleichzeitig stattfindenden Datenübertragungen müssen sich diese Leitung teilen. Dann müssen die Daten durch verschiedene Leitungen durch das Internet übertragen werden, manchmal um die halbe Welt. Diese Leitungen sind zwar sehr leistungsfähig, es gehen aber auch alle Daten des Internet durch diese Leitungen. Eine weiterer Flaschenhals ist die Verbindung zwischen dem Internetanbieter und dem Endkunden, hier ist häufig die grösste Beschränkung der Datenübertragungsrate zu finden (die Datenübertragungsrate ist häufig ein Zehntel oder weniger von der eines Servers). Am Ende müssen die Daten noch vom Router des Endkunden zu seinem Computer gesendet werden. Kabelgebundene Leitungen sind in der Regel schneller als die Anbindungsgeschwindigkeit ans Internet, wenn die Daten aber über W-LAN gesendet werden, kann hier eine weitere Einschränkung sein ... W-LAN schafft häufig nur ein Zehntel der Kabelverbindung.



Webseiten und Stichwörter

Download, Fortschrittsbalken

- <https://de.wikipedia.org/wiki/Fortschrittsbalken>
- https://de.wikipedia.org/wiki/Datenübertragungsrate#Beispiele_für_Datenübertragungsraten





36. Punkte sammeln

Folgendes Denkspiel ist gerade sehr beliebt. Man hat eine Tabelle so wie im Bild unten. Man beginnt im Feld S (für Start) und muss zum Feld Z (für Ziel). Man darf aber in jedem Schritt nur nach rechts und oder oben gehen, wie durch die Pfeile angedeutet. Dabei sammelt man so viele Punkte wie möglich. Die Zahl in jedem Feld gibt an, wie viele Punkte dort gesammelt werden können. Man muss also den Weg so wählen, dass man maximale Punktezahl erreicht.

	2	0	1	1	Z
	1	2	0	2	3
	2	2	0	2	1
	3	1	0	2	0
↑	S	0	1	3	0
				→	

Was ist die maximale Punktezahl, die erreicht werden kann?

- A) 10
- B) 12
- C) 14
- D) 16



Lösung

Die richtige Antwort ist C) 14.

Eine Möglichkeit, die richtige Antwort zu finden, ist die Tabelle mit der jeweils höchsten erreichbaren Zahl auszufüllen. Im Feld unten links starten wir mit 0:

2	0	1	1	Z
1	2	0	2	3
2	2	0	2	1
3	1	0	2	0
0	0	1	3	0

Hierbei ist die fett gedruckte Zahl das Maximum, das wir in jeder Zelle erreichen konnten. Nach oben wird diese Zahl um 3 erhöht, nach rechts um 0:

2	0	1	1	Z
1	2	0	2	3
2	2	0	2	1
3	1	0	2	0
0	0	1	3	0

Das Feld rechts neben der fett gedruckten 3 und oberhalb der fett gedruckten 0 kann nun von unten und von links erreicht werden. Da wir eine möglichst grosse Zahl erreichen wollen, gehen wir von links und erhalten mit der 1 des Feldes also die Summe 4:

2	0	1	1	Z
1	2	0	2	3
2	2	0	2	1
3	4	0	2	0
0	0	1	3	0

Wenn man auf diese Art und Weise weitermacht, kann man für jedes Feld das erreichbare Maximum durch das Maximum der links und darunter liegenden Felder plus dem Wert des Feldes erreichen. Mathematisch ausgedrückt heisst das:

$$v(i, 0) = 0$$

$$v(0, j) = 0$$

$$v(i, j) = c(i, j) + \max\{v(i - 1, j), v(i, j - 1)\}$$

wobei $v(i, j)$ das Maximum des Feldes (i, j) und $c(i, j)$ der ursprüngliche Wert des Feldes (i, j) ist. Da die Formel immer das Feld links und unten verwendet, muss man noch eine Spalte Nullen links und eine Zeile Nullen unten an die Tabelle hinzufügen.

Wenn man die Formel konsequent anwendet, erhält man:

0	8	9	10	12	14
0	6	9	9	11	14
0	5	7	7	9	10
0	3	4	4	6	6
0	0	0	1	4	4
0	0	0	0	0	0



Dies ist Informatik!

Die „beste“ Lösung unter allen möglichen Lösungen zu finden ist teilweise aufwendig und schwer. In diesem konkreten Problem könnte man alle möglichen Wege ausprobieren. Dieses Vorgehen nennt man *brute force*. Leider gibt es häufig viele Wege, in diesem Fall sind es 70 verschiedene Wege! Eine effizientere Lösung wäre, wie in der Answerterklärung verwendet, sich die beste Lösung für Teilschritte zu notieren. Diese Technik heisst *Memoisation* und speichert die Rückgabewerte der Funktion *Dynamischer Programmierung*. Dadurch brauchen nur 25 Werte berechnet werden. In diesem Fall können wir aber einige sich besonders lohnende Teilwege finden, von denen man relativ schnell zur besten Lösung kommen kann. Da die Tabelle recht klein ist, kann man einfach feststellen, dass alle anderen Lösungen schlechter sind.

Webseiten und Stichwörter

Dynamische Programmierung, Memoisation

- https://de.wikipedia.org/wiki/Dynamische_Programmierung
- <https://de.wikipedia.org/wiki/Memoisation>





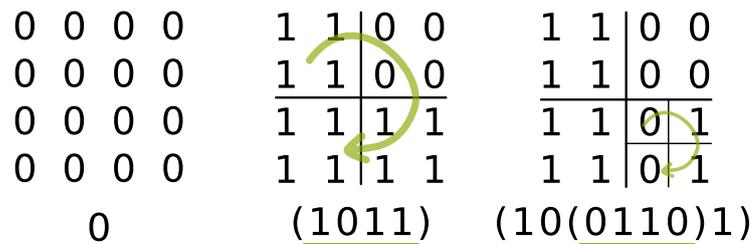
37. Bildviertel

Schwarz-weiße Pixelbilder können mit den Binärzeichen 0 und 1 so dargestellt werden: Eine 0 steht für ein weißes Pixel, eine 1 für ein schwarzes Pixel. Ein Bild mit 4 mal 4 Pixeln wird so mit 16 Zeichen dargestellt.

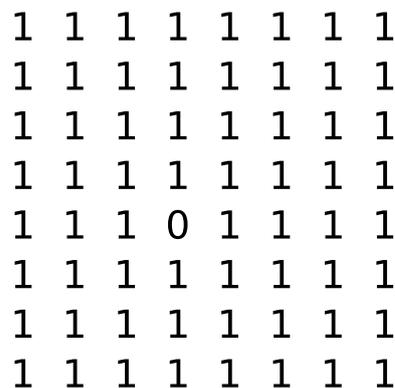
Viele Bilder kann man mit weniger Zeichen darstellen, wenn man das Verfahren *Viertel* anwendet. Dazu werden die Zeichen in einem quadratischen Raster angeordnet. Das Verfahren *Viertel* wird auf ein solches Raster so angewendet:

Falls ein Raster nur aus einem Zeichen besteht, ist das Ergebnis genau dieses Zeichen. Falls alle Zeichen im Raster 0 sind, ist das Ergebnis das Zeichen 0. (s. Bild links). Falls alle Zeichen im Raster 1 sind, ist das Ergebnis das Zeichen 1. Andernfalls wird das Raster in vier gleich große Teil-Raster aufgeteilt.

Das Verfahren *Viertel* wird dann der Reihe nach auf diese Teil-Raster angewendet, von links oben aus im Uhrzeigersinn. Die vier (Teil-)Ergebnisse werden hintereinander geschrieben, zwischen die Klammerzeichen „(“ und „)“ (siehe Bild in der Mitte und rechts). Das Ergebnis ist die so entstandene Zeichenfolge.



Wende das Verfahren *Viertel* auf das folgende 8×8 große Bild. Wie ist das Ergebnis?



- A) (1110)
- B) (11(1011)1)
- C) (111(1(1101)11))
- D) (111(1(1011)11))



Lösung

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
<u>(111(1(1011)11))</u>							

Die Anwendung des Verfahrens Viertel verläuft so (siehe auch das Bild): Zunächst sind nicht alle Zeichen im Raster gleich. Das Raster wird also in vier Raster aufgeteilt. Das Gesamtergebnis wird also so aussehen:

$$(\langle \text{Ergebnis 1} \rangle \langle \text{Ergebnis 2} \rangle \langle \text{Ergebnis 3} \rangle \langle \text{Ergebnis 4} \rangle)$$

In den Teil-Rastern 1, 2 und 3 sind jeweils alle Zeichen gleich 1. Für diese Raster ist das Ergebnis deshalb 1. Das Gesamtergebnis wird also so aussehen:

$$(111 \langle \text{Ergebnis 4} \rangle)$$

Das Teil-Raster 4 muss aufgeteilt werden. In den Teil-Rastern 4.1, 4.3 und 4.4 sind jeweils wieder alle Zeichen gleich 1. Damit wissen wir jetzt, dass das Gesamtergebnis so aussehen wird:

$$(111(1 \langle \text{Ergebnis 4.2} \rangle 11))$$

Teil-Raster 4.2 muss aufgeteilt werden. Dabei entstehen nur einzelne Zeichen. Das Ergebnis für dieses Teil-Raster ist damit (1011). Das Gesamtergebnis ist also:

$$(111(1(1011)11))$$

Dies ist Informatik!

Wenn immer mehr Informatiksysteme wie PCs, Smartphones und Tablets benutzt werden, entstehen immer mehr digitale Daten. Und auch wenn die Systeme immer mehr Kapazität zum Speichern und Übertragen digitaler Daten besitzen: Es reicht nie aus. Es bleibt also interessant, sich zu überlegen, wie man Texte, Bilder, Audio- oder Videodaten komprimieren kann, also so zu speichern, dass man möglichst wenig Speicherplatz braucht und keine Information verloren geht.

Die Informatik kennt viele Verfahren, Daten verlustfrei zu komprimieren. Das in dieser Aufgabe beschriebene Verfahren funktioniert am besten, wenn grosse Bereiche gleiche Daten enthalten. Die Aufteilung in vier gleich grosse Bereiche (und ggf. deren Aufteilung in vier weitere Bereiche usw.) kann man mit Hilfe sogenannter Quadrees im Computer nachvollziehen. Ein Quadtree ist eine besondere Variante der Datenstruktur Baum, die in der Informatik eine besonders wichtige Rolle spielt.

Webseiten und Stichwörter

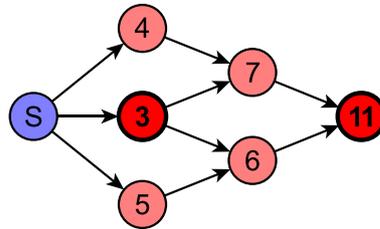
Quadtree, Komprimierung

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



38. Abkürzung oder Umweg?

Das Bild stellt eine Karte mit Einbahnstrassen dar. Die Zahl in den Kreuzungen stellt jeweils die Länge des *kürzesten* Weges von S zu dieser Kreuzung dar.



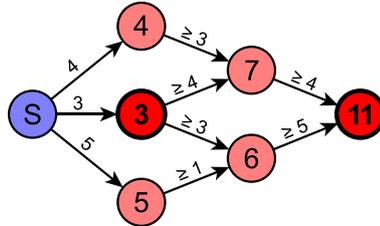
Welche der folgenden Aussagen über die beiden fett markierten Kreuzungen ist immer wahr?

- A) Die Länge des kürzesten Weges zwischen diesen beiden Kreuzungen ist genau 8.
- B) Die Länge des kürzesten Weges zwischen diesen beiden Kreuzungen ist 8 oder weniger als 8.
- C) Die Länge des kürzesten Weges zwischen diesen beiden Kreuzungen ist 8 oder mehr.
- D) Man kann nichts über die Länge des kürzesten Weges zwischen diesen beiden Kreuzungen sagen.



Lösung

Die richtige Antwort ist: „Der kürzeste Weg zwischen diesen beiden Kreuzungen ist 8 oder mehr.“
Wäre der kürzeste Weg weniger als 8, so wäre der kürzeste Weg zwischen S und der Kreuzung mit „11“ weniger als $3 + 8 = 11$. Der kürzeste Weg könnte aber grösser als 8 sein, da der kürzeste Weg zwischen S und der Kreuzung mit „11“ auch über „4“ und „7“ oder über „5“ und „6“ führen könnte. Damit sind die anderen Antworten falsch.



Die Graphik oben zeigt an, wie lang die einzelnen Wegstrecken mindestens sein müssen, damit die angegebenen Zahlen stimmen.

Dies ist Informatik!

Auf den ersten Blick scheint es in dieser Aufgabe lediglich um den kürzesten Weg zwischen zwei Kreuzungen zu gehen. Dafür müsste man aber die tatsächliche Entfernung zwischen zwei Kreuzungen wissen, die man üblicherweise als Weglänge an den Weg schreibt. In diesem Fall jedoch ist nur die Länge des kürzesten Wegs von S zu jeder Kreuzung notiert, es könnte zum Beispiel sein, dass der Weg von „4“ zu „7“ 10 lang ist, wenn der Weg von „3“ zu „7“ 4 lang ist. Daher kann man ohne weitere Informationen nicht sagen, welcher der möglichen Wege der kürzeste Weg ist.

Webseiten und Stichwörter

Kürzester Pfad

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



39. Digitalziffern

Eine 7-Segment-Anzeige stellt die Ziffern auf folgende Weise dar:



Jede der Ziffern besteht aus bis zu 7 Segmenten. Angenommen man hat eine einstellige Ziffernanzeige und die Sicht auf diese Anzeige ist teilweise blockiert.

Welche Segmente dürfen nicht verdeckt sein, damit die Ziffernanzeige noch lesbar ist?





Lösung

Die richtige Antwort ist:



Die Ziffernpaare 1/7, 3/9, 5/6, 6/8 und 8/0 sind der Schlüssel zur Lösung der Aufgabe. Die Ziffernpaare unterscheiden sich nur in einem Segment. Wenn dieses Segment wegfallen würde, könnte man die Ziffern nicht mehr unterscheiden:



Die beiden übrigen Segmente unten rechts sind nicht relevant, um Ziffernpaare zu unterscheiden, da sich alle Ziffernpaare entweder in mehr als diesen beiden Segmenten unterscheiden, oder die Segmente sowieso gleich sind.

Dies ist Informatik!

Wenn man bei einer Information etwas weglassen kann, ohne sie unlesbar zu machen, spricht man von Redundanz. Texte sind ein Beispiel für Redundanz, denn geringfügige Änderungen beeinträchtigen die Lesbarkeit wenig oder gar nicht. Redundanz in Textdateien vergrößert den Platzbedarf, macht die Information darin aber robuster gegen ungewollte Veränderung wie zum Beispiel durch Störungen bei der Speicherung oder Übertragung. Deshalb ist Redundanz grundsätzlich nicht unbedingt ein Problem.

Manchmal will man die Redundanz aus Dateien entfernen. Dieser Vorgang heisst „Datenkompression“. So spart man Speicherplatz oder kann Daten schneller übertragen. Normalerweise wird man einen beschädigten Text meistens dennoch lesen und verstehen können. Ist eine komprimierte Datei jedoch beschädigt, kann man sie meist gar nicht mehr öffnen und schlimmstenfalls ist die ganze Information verloren. Da heutzutage viele Dateien komprimiert sind, gilt dies für viele anderen Dateiarten. Deshalb werden bei komprimierten Dateien manchmal zusätzliche redundante Daten hinzugefügt, mit deren Hilfe man kleine Fehler korrigieren kann.

Webseiten und Stichwörter

Erkennen von Informationen, Minimalisierung, Ziffern, 7-Segment-Anzeige, Redundanz

- <https://de.wikipedia.org/wiki/Segmentanzeige#Siebensegmentanzeige>



40. Zerteile die Ziffernfolge

In einem speziellen Code für Texte wird jeder Buchstabe durch ein Codewort aus den Ziffern 0 bis 9 kodiert. Die Besonderheit: Kein Codewort darf mit dem Codewort eines anderen Buchstabens beginnen.

Der Buchstabe X wird beispielsweise durch 12 kodiert. Nun kann Y durch 2 kodiert werden. Denn 12 beginnt nicht mit 2 (und 2 nicht mit 12). Jetzt kann Z durch 11 kodiert werden; denn weder 12 noch 2 beginnen mit 11, und 11 beginnt weder mit 12 noch mit 2. 21 wäre jedoch nicht als Codewort für Z erlaubt, weil es mit 2, also dem Codewort von Y beginnt.

Das Wort BEBRAS wird durch die Ziffernfolge 12112233321 kodiert. Welche der folgenden Teilungen repräsentiert die Buchstaben?

- A) 12 11 22 33 32 1
- B) 1 21 1 22 33 321
- C) 1 21 12 2 33 321
- D) 1 21 1 22 3 3321
- E) 12 1 12 23 33 21



Lösung

Die richtige Lösung ist:

1 21 1 22 33 321

Wir beginnen links am Anfang der Ziffernfolge. Falls B durch 12 kodiert würde, hätte E zwangsläufig das Codewort 1 (denn dahinter kommt wieder 12 für das zweite B). Das würde jedoch der Präfix-Regel widersprechen. Denn das Codewort für B würde dann mit 1 beginnen, dem Codewort für E. Es ist unmöglich, dass das Codewort für B mehr als 2 Ziffern besitzt (121, 1211, 12112, etc.) weil dieses Codewort zwei Mal vorkommen muss, aber diese Ziffernfolgen jeweils nur einmal in der Ziffernfolge der Aufgabe enthalten sind. Folglich ist das Codewort für B die Ziffer 1.

Nach dem Buchstaben E folgt der Buchstabe B. Somit kann E nur durch folgende Ziffernfolgen kodiert werden: 2, 21 oder 211223332. Es kann nicht 2 sein; denn dann würde Wort mit BEBB beginnen. Es kann nicht 211223332 sein; denn dann wäre das Wort BEB. Folglich ist das Codewort für E die Ziffernfolge 21. Nun wissen wir, dass 1 21 1 die Kodierung für BEB ist. Wir müssen nur noch 2233321 aufteilen.

Betrachten wir den Buchstaben S am Ende des Wortes. Sein Codewort kann weder 1 noch 21 sein, weil diese Ziffernfolgen bereits für B und E verwendet werden. Folglich sind dies die einzig möglichen Codewörter für S: 321, 3321, 33321, 233321 und 2233321. S kann nicht durch 2233321 kodiert werden, weil dann das Wort BEBS wäre. Es kann auch nicht 233321 sein, weil dann nur eine einzige Ziffer für R und A übrig bliebe. Es kann auch nicht 33321 sein, weil dann die beiden gleichen Ziffern in 22 die unterschiedlichen Buchstaben R und A kodieren müssten. Wenn S als 3321 kodiert würde, müssten die Buchstaben RA als 223 kodiert werden. Allerdings kann das Codewort für R nicht 2 sein und das Codewort für A kann nicht nur 3 sein, weil andere Codewörter mit diesen Ziffern beginnen. Folglich muss das Codewort für S die Ziffernfolge 321 sein. Es bleibt die Ziffernfolge 2233, die die Buchstaben RA kodiert. Aus schon genannten Gründen muss R durch 22 und A durch 33 kodiert werden.

Dies ist Informatik!

Der Code, der in dieser Biberaufgabe beschrieben wird, ist ein Beispiel für einen *Präfixcode*. Ein Präfix ist eine Zeichenfolge zu Beginn einer anderen Zeichenfolge. Bei einem Präfixcode darf kein Codewort Präfix eines anderen Codeworts sein. Das heisst: kein Codewort darf mit einem anderen Codewort beginnen.

Bei Präfixcodes haben die Codewörter unterschiedliche Länge. Der Vorteil der Präfix-Regel ist, dass man keine Trennsymbole zwischen Codewörtern benötigt. Man kann immer erkennen, an welcher Stelle das nächste Codewort beginnt. Wenn man kurze Codewörter für häufig vorkommende Buchstaben wählt, kann man Texte sehr effizient kodieren und grosse Textmengen platzsparend speichern. Die Huffman-Kodierung ist eine Methode, einen optimalen Präfixcode zu finden. Sie ist weit verbreitet und steckt z.B. hinter bekannten Datenformaten wie JPEG und MP3.

Webseiten und Stichwörter

Präfixcode, Kryptographie, Kryptoanalyse, Codes knacken

- <https://de.wikipedia.org/wiki/Präfixcode>
- <https://de.wikipedia.org/wiki/Huffman-Kodierung>



A. Aufgabenautoren

Andrea Adamoli	Arnheiður Guðmundsdóttir	J.P. Pretti
Nursultan Akhmetov	Yasemin Gülbahar	Daniel Rakijašić
Adil Aliyev	Martin Guggisberg	Chris Roffey
Haim Averbuch	Urs Hauser	Frances Rosamond
Khuyagbaatar Batsuren	Hans-Werner Hein	Kirsten Schlüter
Wilfried Baumann	Fredrik Heintz	Victor Schmidt
Bartosz Bieganski	Juraj Hromkovič	Eljakim Schrijvers
Daphne Blokhuis	Ungyeol Jung	Masood Seddighin
Eugenio Bravo	Filiz Kalelioğlu	Maiko Shimabuku
Andrej Brodnik	Dong Yoon Kim	Taras Shpot
Carmen Bruni	Vaidotas Kinčius	Martin Stangl
Amaury A. Castro Jr.	Jia-Ling Koh	Seiichi Tani
Marios Choudary	Ivana Kosírová	Gergely Tassy
Anton Chukhnov	Regula Lacher	Monika Tomcsányiová
Raluca Constantinescu	Greg Lee	Peter Tomcsányi
Zsófia Csepregi-Horváth	Milan Lukić	Ahto Truu
Valentina Dagienė	Dario Malchiodi	Willem van der Vegt
Darija Dasović Rakijašić	Hiroki Manabe	Jiří Vaníček
Christian Datzko	Dimitris Mavrovouniotis	Troy Vasiga
Susanne Datzko	Mattia Monga	Nicolette Venn
Janez Demšar	Henry Ong	Corina Elena Vint
Olivier Ens	Zsuzsa Pluhár	Michael Weigend
Hanspeter Erni	Wolfgang Pohl	Hongjin Yeh
Michael Fellows	Ilya Posov	Momo Yokoyama
Gerald Futschek	Sergei Pozdniakov	Khairul A. Mohamad Zaki



B. Sponsoring: Wettbewerb 2017

HASLERSTIFTUNG

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

Stiftungszweck der Hasler Stiftung ist die Förderung der Informations- und Kommunikationstechnologie (IKT) zum Wohl und Nutzen des Denk- und Arbeitsplatzes 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/> & <http://www.galaxus.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.



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

Pädagogische Hochschule Luzern



AUSBILDUNGS- UND BERATUNGSZENTRUM
FÜR INFORMATIKUNTERRICHT

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

Ausbildungs- und Beratungszentrum für Informatikunterricht der ETH Zürich.



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



<https://www.zhdk.ch/>
Zürcher Hochschule der Künste



<http://www.zubler.ch/>
Zubler & Partner AG Informatik
Umfassendes Angebot an Dienstleistungen.



<http://senarclens.com/>
Senarclens Leu & Partner



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

SV!A

www.svia-ssie-ssii.ch
schweizerischer vereinfürinformatikind
erausbildung//sociétésuissedel'inform
atiquedansl'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.