



INFORMATIK-BIBER SCHWEIZ
CASTOR INFORMATIQUE SUISSE
CASTORO INFORMATICO SVIZZERA

Aufgaben und Lösungen 2017

Schuljahre 7/8

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

Herausgeber:
Christian Datzko, Hanspeter Erni

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001

SV!A

www.svia-ssie-ssii.ch
schweizerischerverein für informatik in d
erausbildung // société suisse de l'inform
atique dans l'enseignement // società sviz
zera per l'informatica nell'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. 38 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 Berührungsängste mit dem Schulfach Informatik abbauen und das Interesse an Fragenstellungen dieses Fachs wecken. Der Wettbewerb setzt keine Anwenderkenntnisse im Umgang mit dem Computer voraus – ausser dem „Surfen“ 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>



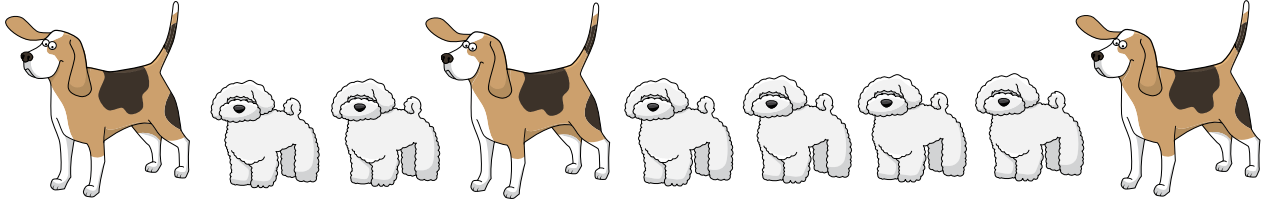
Inhaltsverzeichnis

Mitarbeit Informatik-Biber 2017	i
Vorwort	ii
1. Tauschen von Hunden	1
2. Japanischer Ehrenname	3
3. Kurzes Programm	5
4. Klammeranhänger	7
5. Schülerzeitung	9
6. Honomakato	13
7. Japanischer Stockkampf	17
8. Pizzeria Biberia	19
9. Geheime Bestellung	21
10. Münzdreh	23
11. „Saftladen“	25
12. Alarm im Museum!	29
13. Lichtkunst	31
14. Ersetzungen	33
15. Kugelbahn	35
A. Aufgabenautoren	38
B. Sponsoring: Wettbewerb 2017	39
C. Weiterführende Angebote	42

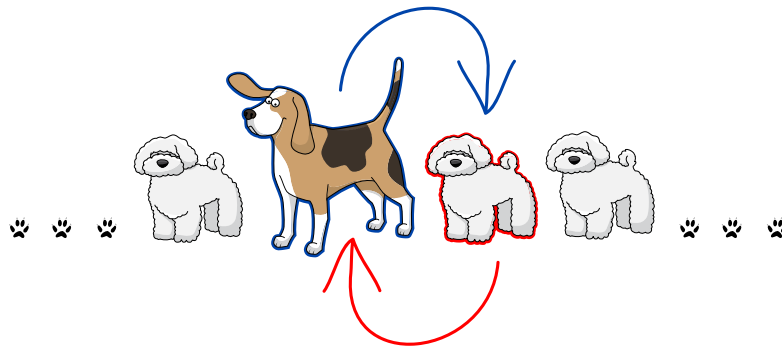


1. Tauschen von Hunden

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>



2. Japanischer Ehrenname

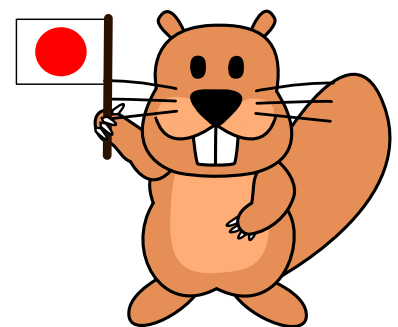
Eine Freundin aus Japan erzählt davon, dass nach einer alten japanischen Tradition aus den Buchstaben des Vornamens Ehrennamen gemacht werden, indem jeder Buchstabe mit einer Silbe ersetzt wird.

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 Jakov. 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 Jakov 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 Jakov 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

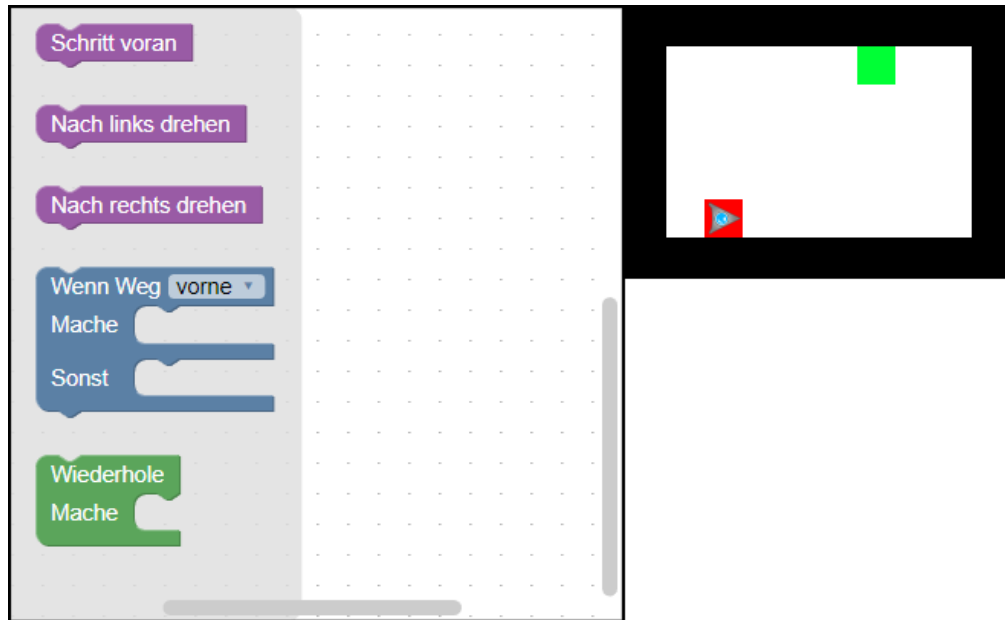
Wortersetzungssystem, Semi-Thue-System, Monoalphabetische Verschlüsselung

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



3. Kurzes Programm

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

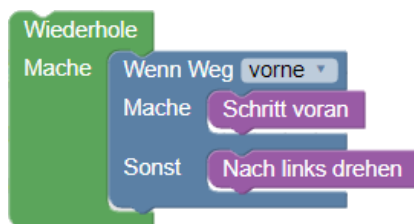


Wähle dazu Programmteile und füge sie auf der Arbeitsfläche zusammen. Du darfst höchstens vier Bausteine für dein Programm benutzen.



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/>

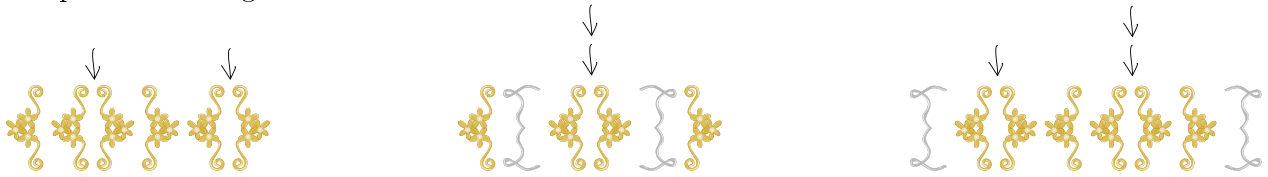


4. Klammeranhänger

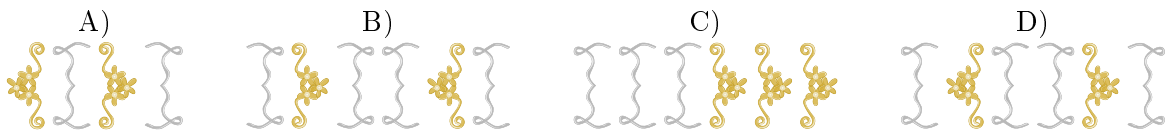
Familie Biber produziert selbstgemachte Anhänger für ein Mittelalterfest. Dazu verwenden sie klammernförmige Ornamente, die immer paarweise verwendet werden. Um einen Anhänger herzustellen, beginnt man mit einem der beiden folgenden Paare:



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>



5. 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		3	3					
Bea			1	1	1	1		
Celine	1	1						
David					3	3	3	
Emma		2	2					
Flo				2	2			
Gioa			5	5		5	5	
Hans		5						
Ida	4	4	4					
Jakob						2	2	

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
1	Celine	Celine	Bea	Bea	Bea	Bea		
2		Emma	Emma	Flo	Flo	Jakob	Jakob	
3		Anna	Anna		David	David	David	
4	Ida	Ida	Ida					
5		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

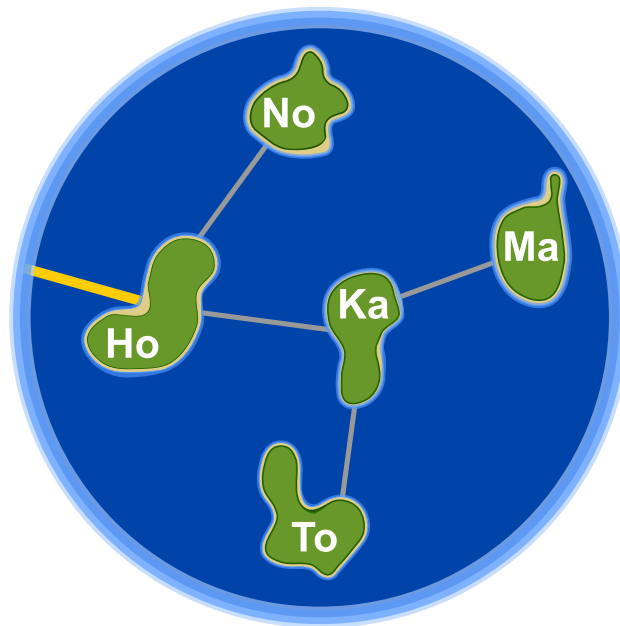
- <https://de.wikipedia.org/wiki/Intervallgraph>





6. 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.

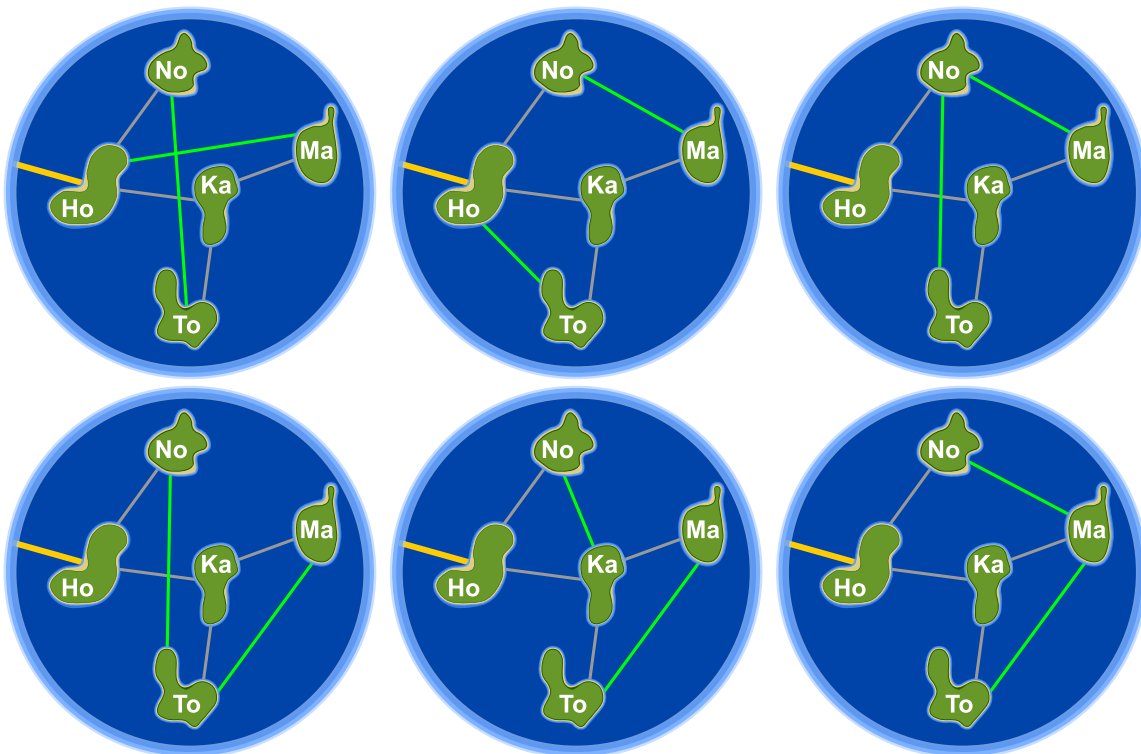


Lösung

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

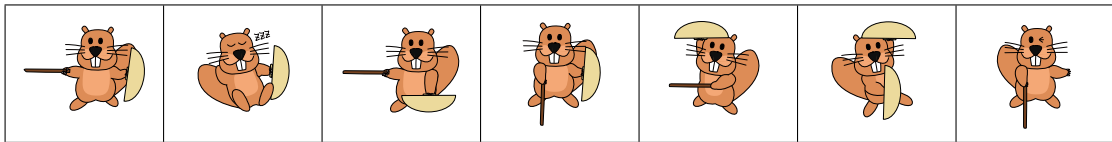
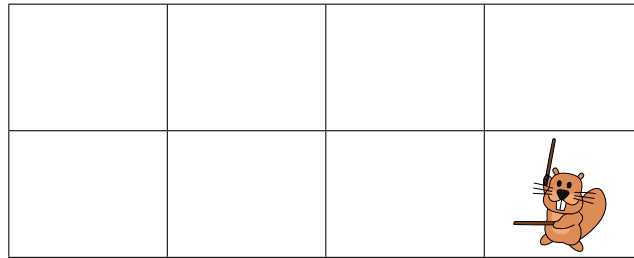
- [https://de.wikipedia.org/wiki/Trenner_\(Graphentheorie\)](https://de.wikipedia.org/wiki/Trenner_(Graphentheorie))





7. 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:

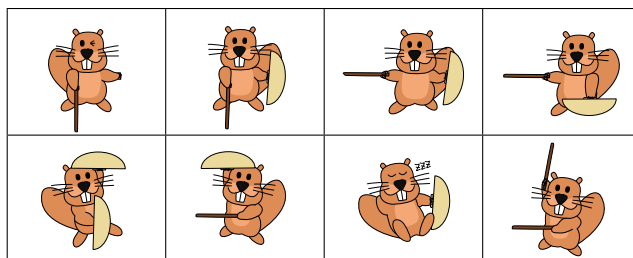


Ordne die Bilder der Freunde den Feldern auf dem Schulhof zu, so dass am Ende jeder Stock auf ein Schild zeigt.



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

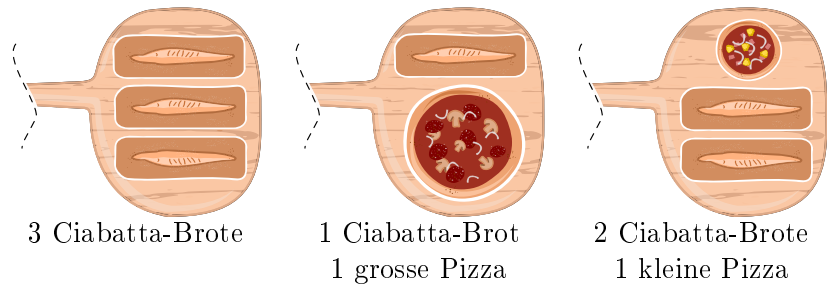
- <https://de.wikipedia.org/wiki/Backtracking>
- http://www.inf-schule.de/grenzen/komplexitaet/affenpuzzle/einstieg_affenpuzzle



8. 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.

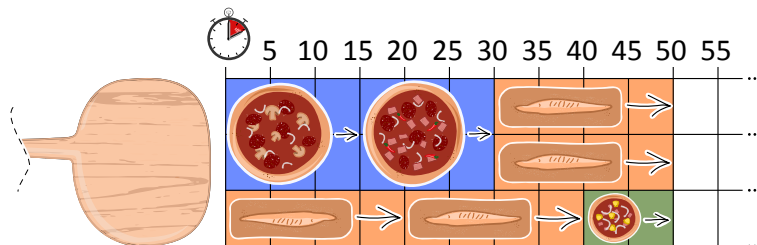
In wie vielen Minuten kann der Pizzabäcker die Bestellung schnellstmöglich backen lassen?





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 backt 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.

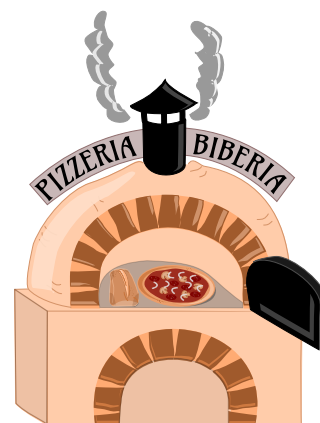
Dies ist Informatik!

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

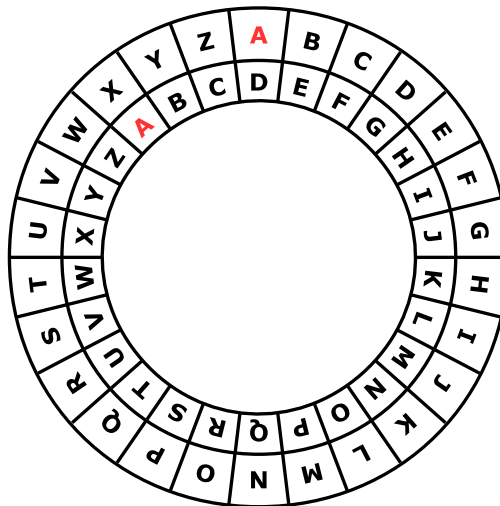
Schedule, Warteschlange, Ressource, Round-Robin, Greedy

- <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>



9. 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 (aussen), 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

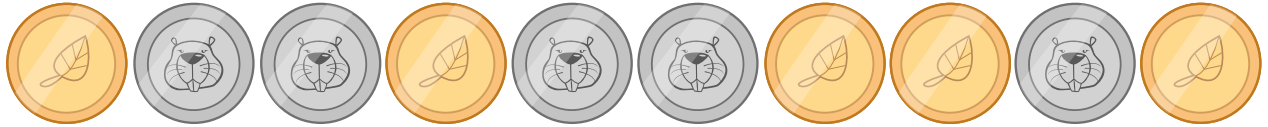
- <https://de.wikipedia.org/wiki/Caesar-Verschlüsselung>
- https://de.wikipedia.org/wiki/Polyalphabetische_Substitution



10. 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








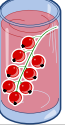



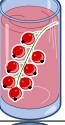



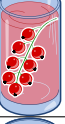
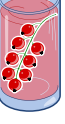



Parität, Paritätsbit

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



11. „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

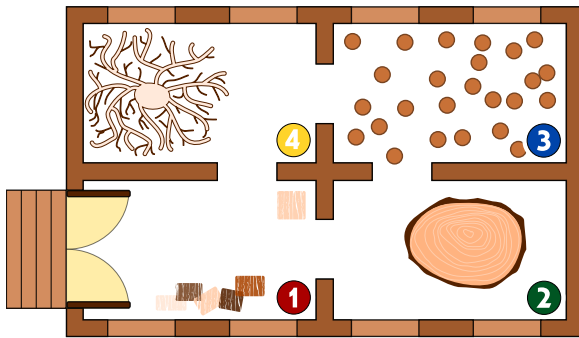
Optimierung, Matching

- [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))





12. 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

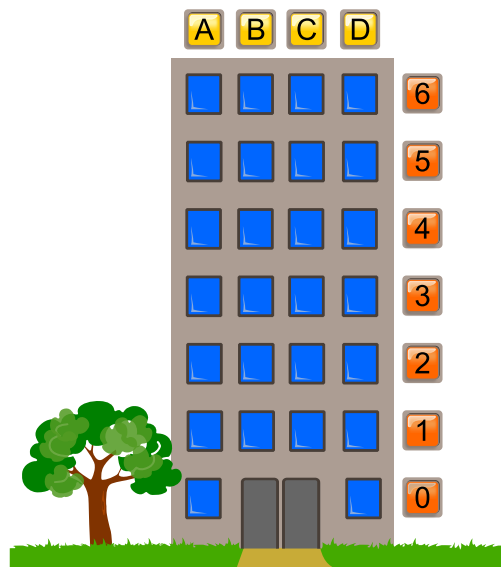
Intelligentes Sicherheitssystem, Regel, Zustandsübergang

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

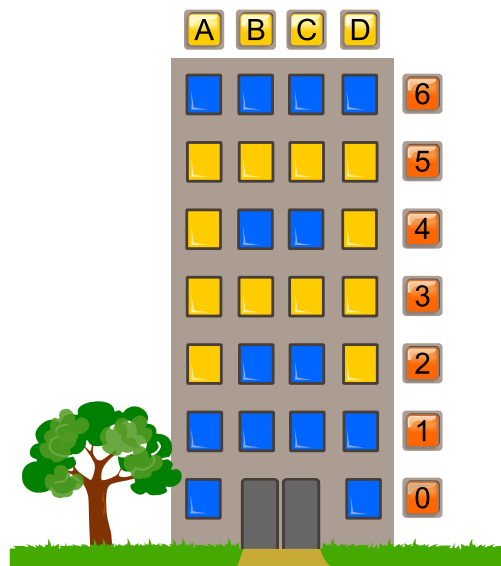


13. Lichtkunst

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>



14. 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

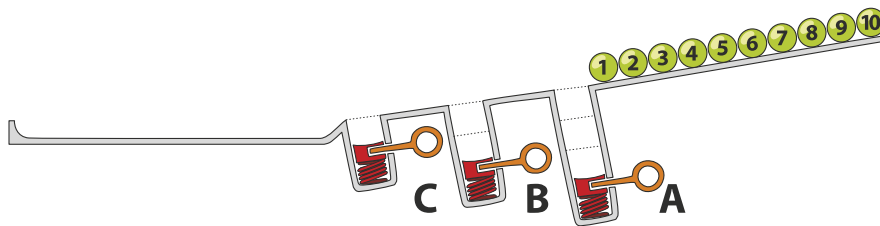
- https://de.wikipedia.org/wiki/Formale_Grammatik



15. 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?

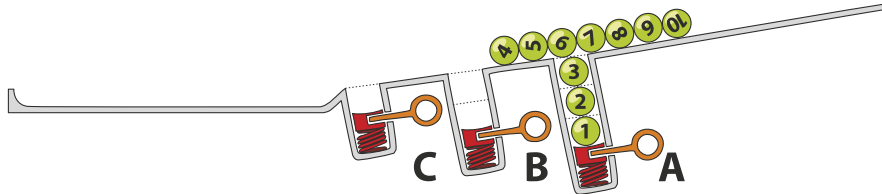
- A)
- B)
- C)
- D)



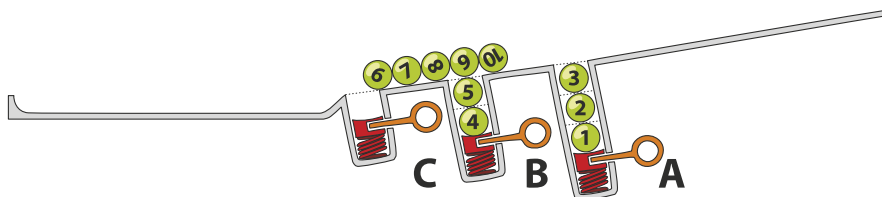
Lösung

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

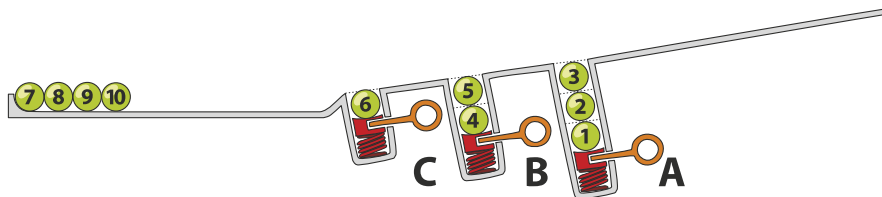
Die Kugeln 1, 2 und 3 fallen in Loch A, die Kugeln 4 bis 10 rollen über sie hinweg.



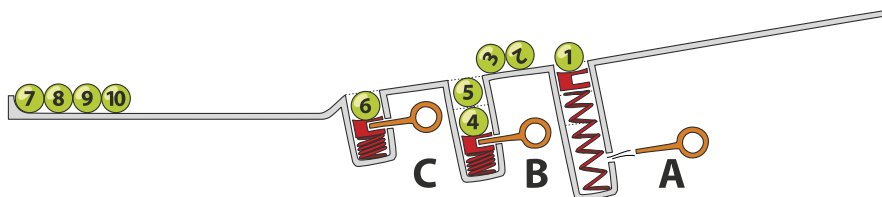
Kugeln 4 und 5 fallen in Loch B und Kugeln 6 bis 10 rollen über sie hinweg.



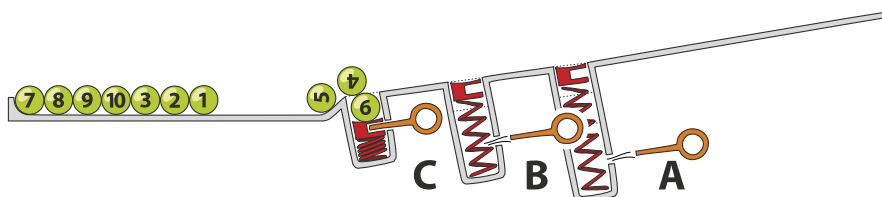
Schliesslich fällt Kugel 6 in Loch C und die Kugeln 7 bis 10 rollen in dieser Reihenfolge bis an das Ende der Rampe.



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.

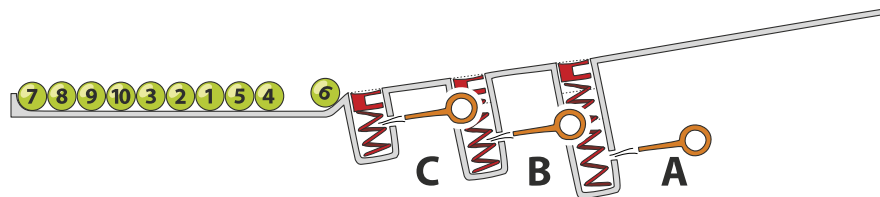


Nun sind die Kugeln in der Reihenfolge 7, 8, 9, 10, 3, 2, 1 am Ende der Rampe. Dann wird die Feder in Loch B gelöst, so dass 5 und 4 an das Ende rollen und zuletzt wird Kugel 6 aus Loch C hinausgeschoben.





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>



A. Aufgabenautoren

 Andrea Adamoli
 Wilfried Baumann
 Daphne Blokhuis
 Eugenio Bravo
 Andrej Brodnik
 Carmen Bruni
 Amaury A. Castro Jr.
 Anton Chukhnov
 Raluca Constantinescu
 Valentina Dagiene
 Christian Datzko
 Susanne Datzko
 Janez Demšar
 Olivier Ens
 Hanspeter Erni

 Gerald Futschek
 Martin Guggisberg
 Urs Hauser
 Hans-Werner Hein
 Juraj Hromkovič
 Ungeol Jung
 Ivana Kosírová
 Regula Lacher
 Greg Lee
 Milan Lukić
 Dario Malchiodi
 Hiroki Manabe
 Mattia Monga
 Zsuzsa Pluhár
 Wolfgang Pohl

 J.P. Pretti
 Daniel Rakijašić
 Chris Roffey
 Frances Rosamond
 Kirsten Schlüter
 Eljakim Schrijvers
 Maiko Shimabuku
 Taras Shpot
 Martin Stangl
 Gergely Tassy
 Ahto Truu
 Corina Elena Vint
 Michael Weigend
 Hongjin Yeh
 Momo Yokoyama



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 Werkplatzes Schweiz. Die Stiftung will aktiv dazu beitragen, dass die Schweiz in Wissenschaft und Technologie auch in Zukunft eine führende Stellung innehat.



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

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

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

RoboRobo ist genial und ermöglicht ein gemeinsames Lern-Erlebnis!



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



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

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

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

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