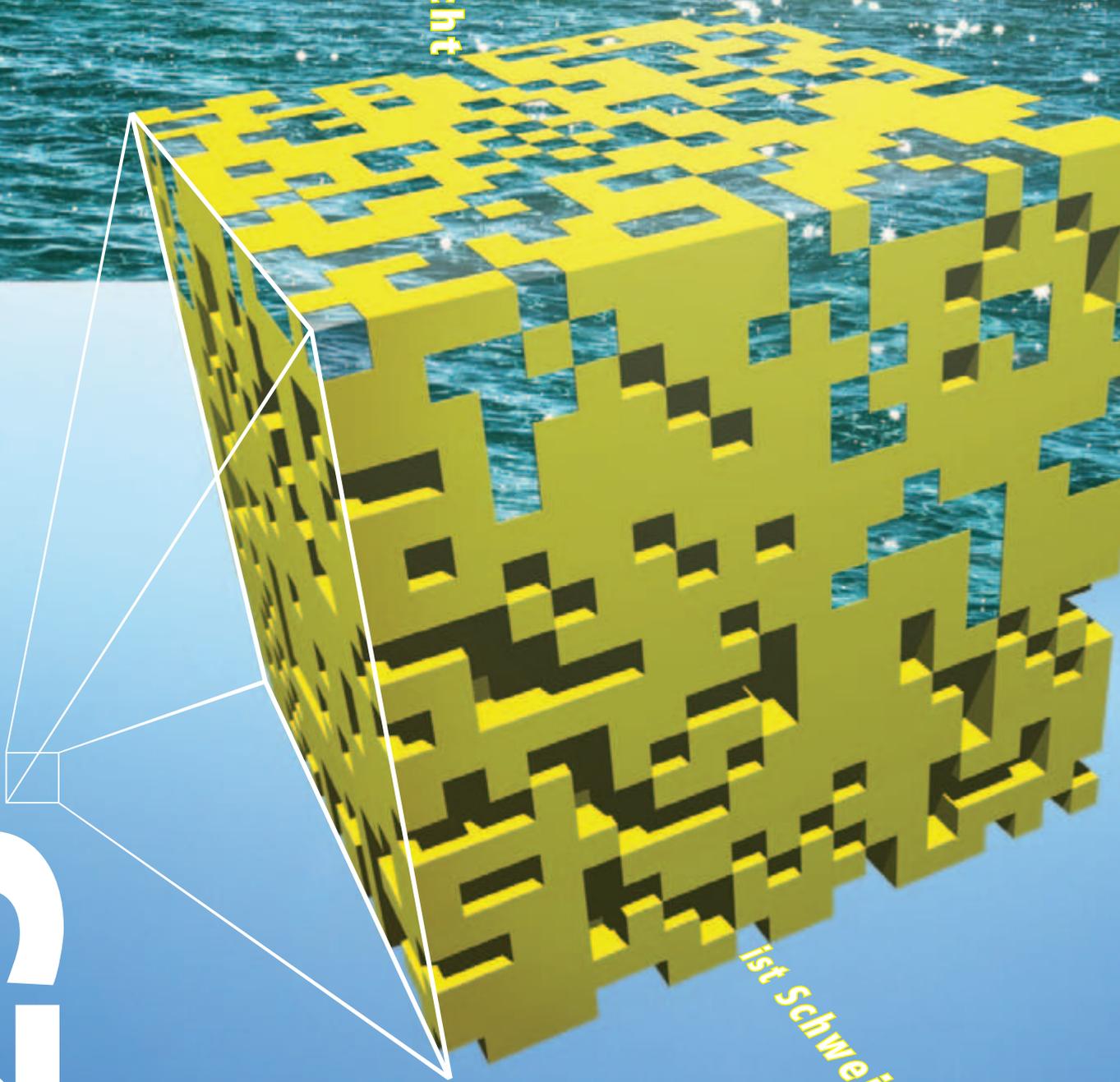


Wie wasserdicht



ist Schweizer Käse ?

23



GEFÖRDERT VOM



Bundesministerium  
für Bildung  
und Forschung

**Bundeswettbewerb Informatik**

**Das Aufgabenblatt**

## Grußwort



Was hat Schweizer Käse mit Informatik zu tun? Zunächst einmal Phantasie, um beides mit der Vorstellung von wasserdurchlässigen Würfeln zu verbinden. Und zum Zweiten die Kreativität, um der Frage nachzugehen, ob und wenn ja, wie wasserdicht ein Würfel ist.

Phantasie und Kreativität gehören zu den wichtigsten Eigenschaften, die für eine Teilnahme am Bundeswettbewerb Informatik notwendig sind. Ich freue mich darüber, dass es dem Aufgabenausschuss in jedem Jahr gelingt, mit seinen Aufgaben die Neugierde zu wecken, die es braucht, um diese Eigenschaften besonders gut zum Tragen zu bringen.

Den PC und das Internet nutzt inzwischen die Mehrzahl der Menschen in unserem Land – nicht nur, aber vor allem die junge Generation. Vergleichsweise wenige jedoch dringen bis zur intensiven Beschäftigung mit Hardware-Systemen oder Software Engineering vor. Dabei stecken hinter Stichworten wie „Mensch-Maschine-Interaktion“ und „Virtuelle und Erweiterte Realität“ Technologien, denen die Zukunft gehört. Und Software Engineering ist bereits auf dem besten Wege, zur „Produktionstechnik des 21. Jahrhunderts“ zu werden.

Wie finden wir einfache Systemlösungen, die es dem Menschen erlauben, durch Sprache, Gesten, Greif- oder Druckbewegungen technische Geräte zu steuern? Welche neuen Software-Systeme brauchen wir, um in der internationalen Entwicklung mithalten zu können? Wie verbessern wir die Basistechnologien für die Kommunikationstechnik, um den immer höheren Nutzungsgraden und komplexeren Strukturen gerecht zu werden?

Das Bundesministerium für Bildung und Forschung unterstützt die Forschung zu all diesen und zahlreichen anderen wichtigen Fragen wesentlich durch das Forschungsprogramm „Informationstechnik 2006“. Denn die Informatik ist nicht nur eine sehr spannende natur- und ingenieurwissenschaftliche Disziplin, sondern auch eine besonders zukunftsstrahlende.

Deshalb ist es mir sehr wichtig, gerade junge Menschen wesentlich stärker als bisher für die Informatik zu interessieren. Das Wissenschaftsjahr 2004 steht auch aus diesem Grund unter dem Motto „Jahr der Technik“ und trägt mit einer nennenswerten Zahl von Veranstaltungen gezielt dazu bei, auch über die IuK-Technologien und ihre vielfältigen Anwendungsbereiche zu informieren und wichtige Forschungsergebnisse für eine breite Öffentlichkeit zugänglich zu machen. Ebenso ein wichtiges, weil stetes, Element der Nachwuchsförderung ist der Bundeswettbewerb Informatik. Er spornt zur vertieften Auseinandersetzung mit dem Fachgebiet an und hilft, die eigene Begabung, die eigenen Fähig- und Fertigkeiten zu erkennen und weiter zu entwickeln. Bei der Beantwortung der diesjährigen Schlüsselfrage „Wie wasserdicht ist Schweizer Käse?“ hoffe ich darum auf möglichst viele Teilnehmerinnen und Teilnehmer und wünsche allen eine kreative Lösungssuche und viel Erfolg.

Edelgard Bulmahn,  
Bundesministerin für Bildung und Forschung

## Gesellschaft für Informatik e.V.

Die Gesellschaft für Informatik e.V. (GI) ist mit rund 24.000 Mitgliedern die größte Fachgesellschaft im Bereich Informatik. Ihre Mitglieder kommen aus allen Sparten der Wissenschaft, der Informatikindustrie, aus dem Kreis der Anwender sowie aus Lehre, Studium und Ausbildung. In der GI wirken Männer und Frauen am Fortschritt der Informatik mit, vor allem im wissenschaftlich-fachlichen Austausch in etwa 120 verschiedenen Fachgruppen und 35 Regionalgruppen. Ihr gemeinsames Ziel ist die Förderung der Informatik in Forschung, Lehre und Anwendung, die gegenseitige Unterstützung bei der Arbeit sowie die Weiterbildung. Die GI vertritt hierbei die Interessen der Informatik in Politik und Wirtschaft. Im Web: [www.gi-ev.de](http://www.gi-ev.de)

## Fraunhofer-Gesellschaft, IuK-Gruppe

Als größter europäischer Forschungsverbund für Informations- und Kommunikationstechnik (IuK) versteht sich die Fraunhofer-IuK-Gruppe als Anlaufstelle für Industriekunden auf der Suche nach dem richtigen Ansprechpartner in der anwendungsorientierten Forschung. Die Vernetzung von insgesamt 3000 Mitarbeitern an bundesweit 17 Forschungsinstituten ermöglicht anbieterunabhängige Technologieberatung und branchenspezifische IT-Lösungen, oft zusammen mit Partnern aus der Industrie. Entwickelt werden IuK-Lösungen für die Geschäftsfelder Digitale Medien, E-Business, EGovernment, Kommunikationssysteme, Kultur und Unterhaltung, Medizin und Life Sciences, Produktion, Security, Software Engineering sowie Verkehr und Mobilität. Weitere Informationen bei der Geschäftsstelle der IuK-Gruppe: [www.iuk.fhg.de](http://www.iuk.fhg.de).



Bundesministerium  
für Bildung  
und Forschung



Gesellschaft  
für Informatik  
e.V.



Fraunhofer  
Gruppe  
Informations- und  
Kommunikationstechnik

Unter der Schirmherrschaft des Bundespräsidenten

# Bundeswettbewerb Informatik

Der Bundeswettbewerb Informatik wurde 1980 von der Gesellschaft für Informatik e.V. (GI) auf Initiative von Prof. Dr. Volker Claus ins Leben gerufen. Ziel des Wettbewerbs ist es, Interesse an der Informatik zu wecken und zu intensiver Beschäftigung mit ihren Inhalten und Methoden sowie den Perspektiven ihrer Anwendung anzuregen. Er gehört zu den bundesweiten Schülerwettbewerben, die von den Kultusministern der Länder unterstützt werden. Gefördert wird er vom Bundesministerium für Bildung und Forschung und steht unter der Schirmherrschaft des Bundespräsidenten. Die Träger des Wettbewerbs sind die GI und die Fraunhofer-Gesellschaft. Die Gestaltung des Wettbewerbs und die Auswahl der Sieger obliegen dem Beirat; Vorsitzender: Prof. Dr. Uwe Schöning, Universität Ulm. Die Auswahl und Entwicklung von Aufgaben und die Festlegung von Bewertungsverfahren übernimmt der Aufgabenausschuss. Die Geschäftsstelle des Wettbewerbs ist in Bonn und ist für die fachliche und organisatorische Durchführung zuständig; Geschäftsführer: Dr. Wolfgang Pohl.

## → Start und Ziel im September

Der Wettbewerb beginnt und endet im September, dauert etwa ein Jahr und besteht aus drei Runden. In der ersten und zweiten Runde sind fünf bzw. drei Aufgaben zu Hause selbstständig zu bearbeiten. Dabei können die Aufgaben der ersten Runde mit grundlegenden Informatikkenntnissen gelöst werden; die Aufgaben der zweiten Runde sind deutlich schwieriger. In der ersten Runde ist Gruppenarbeit zugelassen und erwünscht. An der zweiten Runde dürfen jene teilnehmen, die allein oder zusammen mit anderen wenigstens drei Aufgaben weitgehend richtig gelöst haben. In der zweiten Runde ist dann eigenständige Einzelarbeit gefordert; die Bewertung erfolgt durch eine relative Platzierung der Arbeiten. Die ca. dreißig bundesweit Besten werden zur dritten Runde, einem Kolloquium, eingeladen. Darin führt jeder ein Gespräch mit je einem Informatiker aus Schule und Hochschule und analysiert und bearbeitet im Team zwei Informatik-Probleme.

## → Wer ist teilnahmeberechtigt?

Teilnehmen können Jugendliche, die nach dem 15.11.1982 geboren wurden. Sie dürfen jedoch zum 1.9.2004 noch nicht ihre (informatikbezogene) Ausbildung abgeschlossen oder eine Berufstätigkeit aufgenommen haben. Ebenfalls ausgeschlossen sind Personen, die zum Wintersemester 2004/2005 oder früher ihr Studium an einer Hochschule/Fachhochschule aufnehmen bzw. aufgenommen haben. Jugendliche, die nicht deutsche Staatsangehörige sind, müssen wenigstens vom 1.9. bis 15.11.2004 ihren Wohnsitz in Deutschland haben oder eine staatlich anerkannte deutsche Schule im Ausland besuchen.

## → Junioraufgabe

Um die Teilnahme jüngerer Schülerinnen und Schüler am BWINF zu fördern, wird in diesem Wettbewerb zum ersten Mal eine „Junioraufgabe“ gestellt. Sie darf von bis zu 16-Jährigen bearbeitet werden (geboren nach dem 15.11.1987) bzw. durch Gruppen mit einem solchen Mitglied.

## → Als Anerkennung ...

In allen Runden des Wettbewerbs wird die Teilnahme durch eine Urkunde bestätigt. In der ersten Runde werden darüber hinaus erste und zweite Preise sowie Anerkennungen vergeben; mit einem Preis ist die Qualifikation für die zweite Runde verbunden. Auch in der zweiten Runde gibt es erste und zweite Preise; jüngere Teilnehmer haben die Chance auf eine Einladung zu einer Schülerakademie. Die Gewinner eines ersten Preises in der zweiten Runde werden zur dritten Runde eingeladen, die im Herbst 2005 vom Hasso-Plattner-Institut für Softwaresystemtechnik in Potsdam mit Unterstützung der SAP AG ausgerichtet werden wird. Die dort ermittelten Bundessieger werden in der Regel ohne weiteres Aufnahmeverfahren in die Studienstiftung des deutschen Volkes aufgenommen. Zusätzlich sind für den Bundessieger, aber auch für andere besondere Leistungen Geld- und Sachpreise vorgesehen.

## → ... Teilnahme an der Informatik-Olympiade

Ausgewählte Endrundenteilnehmer können sich in mehreren Trainingsrunden für das vierköpfige deutsche Team qualifizieren, das an der Internationalen Informatik-Olympiade 2006 in Mexiko teilnimmt.

## →



O'REILLY®



Die Sponsoren wünschen allen Teilnehmern des 23. Bundeswettbewerbs Informatik viel Erfolg!

## → ... Preise für Teilnehmer, Lehrer und Schulen

Auch beim 23. BWINF gibt es wieder einiges zu gewinnen. 10 aktuelle Versionen von SUSE Linux werden vergeben, und zwar u.a. an die besten Errundenteilnehmer sowie an Lehrkräfte und Schulen mit besonders vielen Teilnehmern. Eine Chance auf ein SUSE-Paket haben auch diejenigen, die schon mal beim BWINF mitgemacht haben, beim 23. BWINF wieder dabei sind und eine Person ohne BWINF-Erfahrung für die Teilnahme am 23. BWINF werben konnten. Der oder die Geworbene muss dazu Namen und Geburtsdatum des/der Werbenden bei der Anmeldung angeben. In der zweiten Runde vergibt der Verlag O'Reilly erneut Buchpreise.

## → ... erstmals Sonderpreis für Auszubildende

Schon seit einigen Jahren können innerhalb der Altersbegrenzung auch Auszubildende am BWINF teilnehmen. In diesem Jahr wird ganz ausdrücklich in den Ausbildungsbetrieben dazu aufgerufen, denn erstmals stiftet die Gesellschaft für Informatik im 23. BWINF einen Sonderpreis. Als Gewinn winken attraktive Sachpreise, gestiftet u.a. von der Firma DATEV.

## Aufgabe 1

### tratsCH trATsch

Die Leute vom Planeten Chator schreiben gern Schlechtes übereinander. Wer vielen über andere Schlechtes schreibt, gilt als besonders charmant. Aber natürlich nur, wenn die Kompromittierten nichts davon erfahren.

Chatonen schreiben nur an Leute, die Ihnen sympathisch sind. Doch die können den Tratsch weitertragen, und eventuell genau an den Falschen. Ein Chatone muss also gut aufpassen, dass er keinen Charmefehler macht. Dieses Missgeschick passierte unlängst Ator, als er Btor Schlechtes über Dtor schrieb. Zu dumm: Dtor ist dem Ctor sympathisch, der wiederum Btor sympathisch ist. Und so landete der Tratsch bei Dtor, der über Ator verständlicherweise sehr verärgert war. Dies hätte Ator mit ein wenig Übersicht vermeiden können, denn schließlich wissen alle Chatonen voneinander, wer wem sympathisch ist.

#### → Aufgabe

Programmiere einen Charminator, der einliest, welche Chatonen welchen anderen sympathisch sind. Er soll auf dieser Grundlage möglichst kompakt für alle Chatonen ausgeben, wem der Betreffende über wen Schlechtes schreiben kann, ohne einen Charmefehler zu riskieren.

Teste dein Programm an drei Beispielen für mindestens 5 und höchstens 10 Chatonen; verwende auf jeden Fall das folgende Beispiel:

Dem Chatonen	sind sympathisch:
A	B E
B	C
C	D G
D	C
E	F
F	B E G
G	H
H	D

Bedenke dabei, dass eine Tratschnachricht von Chatone x nach Chatone y weder x noch y betrifft und dass ein Chatone eine Nachricht, die ursprünglich von ihm selbst stammt und wieder bei ihm ankommt, nicht weiterleiten wird – so dumm sind Chatonen nicht.

## Aufgabe 2

### Kosmische Schaltjahre

Im 32. Jahrhundert, nach der Entdeckung der Bio-Quanten-Teleportation, besiedeln Menschen immer mehr unbewohnte Planeten des Universums. Jeder Planet X dreht sich in einer Zeit  $t$  um seine eigene Achse und umkreist gleichzeitig in einer Zeit  $a \cdot t$  den zentralen Stern seines Planetensystems. Dadurch ergibt sich ein X-Jahr bestehend aus  $a$  X-Tagen, wobei  $a$  eine positive reelle Zahl ist, die meistens größer als 100 ist. Aus sentimentalen Gründen möchten die Bewohner des Planeten X ihr X-Jahr in X-Monate einteilen, von denen jeder aus einer ganzen Anzahl von X-Tagen besteht, und zwar aus ungefähr 30 X-Tagen. Wenn  $a$  nicht zufällig eine ganze Zahl ist, muss außerdem für den Kalender des Planeten X eine Schaltjahr-Regelung vereinbart werden.

#### → Beispiel:

Wenn das X-Jahr aus  $a=250,26192$  X-Tagen besteht, kann man das X-Jahr in 5 X-Monate von 31 Tagen, 2 X-Monate von 32 Tagen und einen Monat einteilen, der in Schaltjahren aus 32 Tagen, sonst aber aus 31 Tagen besteht. Schaltjahre sind die Jahre, deren Nummer durch 4 oder 7, nicht aber durch 6 teilbar sind. Z.B. ist das Jahr 21 (nach der Besiedelung) ein Schaltjahr, das Jahr 24 aber nicht.

#### → Aufgabe

Schreibe ein Programm, das für einen Planeten X die Länge  $a$  des X-Jahres in X-Tagen einliest und einen möglichst sinnvollen Vorschlag für die Einteilung des X-Jahres in X-Monate sowie für eine dazu passende Schaltjahr-Regelung ausgibt. Das Programm soll auch schätzen, nach wievielen X-Jahren der vorgeschlagene Kalender um einen X-Tag von den astronomischen Realitäten abweicht. Beschreibe und begründe die Wirkungsweise deines Programms und zeige seine Ausgabe bei 5 verschiedenen Werten von  $a$ , darunter 416,78132 und 365,24219.

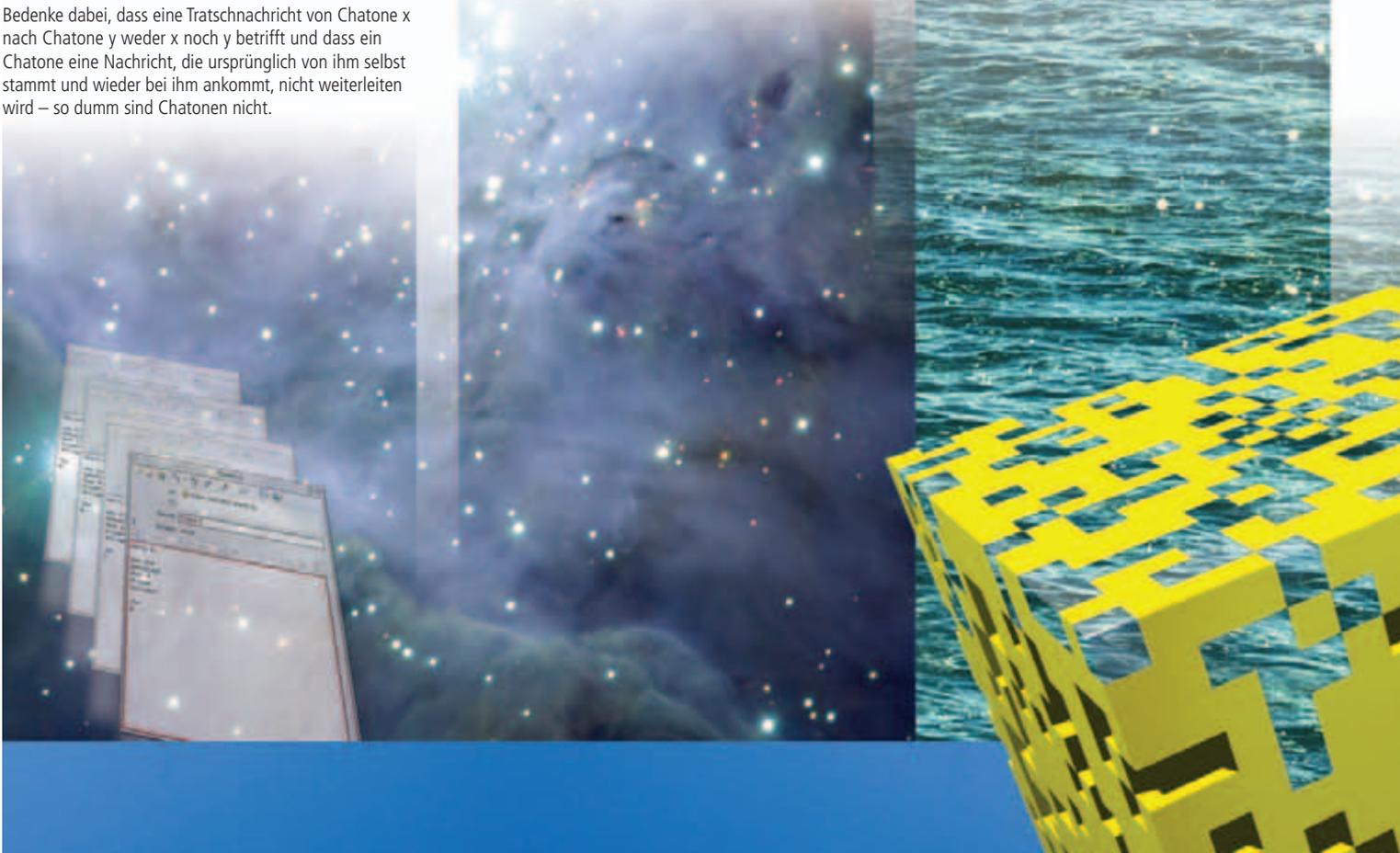
## Aufgabe 3

### Wie wasserdicht ist Schweizer Käse?

Stellen wir uns ein würfelförmiges Stück Schweizer Käse vor. Wir modellieren den Käse durch ein  $20 \times 20 \times 20$ -Raster. Vereinfachend gehen wir davon aus, dass jeder Rasterwürfel ganz aus Käse oder ganz aus Luft besteht. Auf die obere Seite unseres Schweizer Käsewürfels wird nun Wasser gegossen, welches dann nur durch die Luftlöcher in den Würfel eindringt und ihn möglicherweise auf einem durchgängigen Kanal von der Oberseite bis zur Unterseite durchfließt. Dabei kann von einem Luftwürfel zum nächsten nur dann Wasser fließen, wenn die beiden Luftwürfel eine ganze Seite gemeinsam haben (nicht nur eine Kante oder eine Ecke). Das Wasser kann auch auf „Umwegen“ fließen wie z.B. beim Siphon eines Waschbeckenabflusses.

#### → Aufgabe

1. Schreibe ein Programm, das einen Würfel mit einer zufälligen Verteilung von Luft- und Käsewürfeln erzeugt, bei der die Wahrscheinlichkeit für Käse  $p$  und für Luft  $1-p$  beträgt, und prüft, ob aus dem Würfel unten Wasser herauskommt.
2. Untersuche mit Hilfe von Simulationen den Zusammenhang zwischen  $p$  und der Durchflusswahrscheinlichkeit. Stelle die Ergebnisse anschaulich dar und interpretiere sie.



Wilfried H., Lehrer an der Sapi-Schule, stellt seine Klassenarbeiten, indem er zufällig Aufgaben aus einem Fundus auswählt. Die Schüler haben im Laufe der Jahre einige Klassenarbeiten gesammelt. Gaby hat herausgefunden, dass die Aufgaben aus einer Buchreihe stammen, die noch im Buchhandel erhältlich ist. Doch selbst in dieser Buchreihe wurden Aufgaben wiederverwendet, so dass eine Aufgabe in mehreren Bänden enthalten sein kann. Jeder Band der Buchreihe kostet 10 Euro. Die Schüler der 10b wollen sich für die nächste Klassenarbeit „optimal“ vorbereiten und daher alle möglichen Aufgaben vorher üben. Die Bände der Buchreihe enthalten folgende Aufgaben:

<b>Band I</b>	enthält die Aufgaben	<b>1, 9, 12, 15, 18</b>
<b>Band II</b>	enthält die Aufgaben	<b>2, 3, 5, 14, 15</b>
<b>Band III</b>	enthält die Aufgaben	<b>4, 7, 13, 16, 18</b>
<b>Band IV</b>	enthält die Aufgaben	<b>5, 11, 16, 19, 20</b>
<b>Band V</b>	enthält die Aufgaben	<b>6, 8, 11, 14, 15</b>
<b>Band VI</b>	enthält die Aufgaben	<b>1, 3, 6, 17, 20</b>
<b>Band VII</b>	enthält die Aufgaben	<b>2, 8, 10, 13, 19</b>
<b>Band VIII</b>	enthält die Aufgaben	<b>4, 9, 10, 12, 16</b>

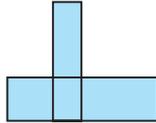
Die folgenden Aufgaben wurden bereits von älteren Mitschülern gesammelt:  
**1, 2, 4, 6, 7, 8, 10, 12, 14, 16, 18, 20**

Gaby bekommt den Auftrag, alle Aufgaben zu besorgen, die noch nicht gesammelt wurden. Sie soll natürlich sparen und versucht daher, möglichst wenig Bände zu kaufen.

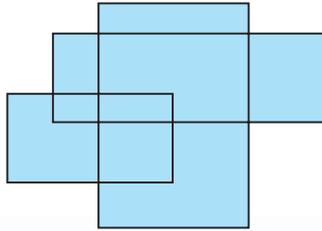
→ **Aufgabe**

Schreibe ein Programm, das Gaby hilft, die richtigen Bände auszuwählen. Aber auch für Schüler anderer Schulen mit anderen Buchreihen, Bänden, Aufgabenzusammenstellungen und bestehenden Sammlungen soll das Programm nützlich sein. Demonstriere seine Fähigkeiten anhand des obigen und zweier weiterer Beispiele.

Frank Zapper ist wieder einmal bei dem Fernsehsender Abzocker-13 gelandet, bei dem man vielleicht auf die Schnelle Geld machen kann, wenn man über eine teure Telefonnummer anruft und eine Aufgabe richtig löst. Die nette Moderatorin zeigt das folgende Bild und fragt nach der Anzahl der Rechtecke, die man hier insgesamt sehen kann.



Das ist aber einfach, denkt Frank Zapper, es sind doch genau 8 Rechtecke. Er will schon anrufen. Da kommt das nächste Bild. Das ist schon schwieriger. Sieht man jetzt 15, 18, 19 oder sogar 20 Rechtecke?



Um auf weitere derartige Aufgaben vorbereitet zu sein, will Frank seinen Computer einsetzen. Du kannst ihm helfen.

→ **Aufgabe**

- Gegeben ist ein rechteckiges Raster der Größe 30x20. In dieses Raster sollen mehrere Rechtecke gezeichnet werden, die jeweils durch ganzzahlige Koordinaten für die linke untere Ecke (x1;y1) und die rechte obere Ecke (x2;y2) gekennzeichnet sind. Erzeuge jeweils einen Datensatz für Mengen von drei, vier und fünf Rechtecken und zeichne für diese Datensätze jeweils das Raster mit den Rechtecken.
- Schreibe ein Programm, das für eine Menge von Rechtecken im Raster die sich insgesamt ergebenden Rechtecke bestimmt und zählt. Dokumentiere die Programmresultate durch die Rechtecklisten und Rechteckanzahlen für deine in Teilaufgabe 1 erzeugten Rechteckmengen und die beiden Mengen

R1 = {(3;2), (13;5)}, {(6;2), (8;10)} und  
 R2 = {(2;6), (13;12)}, {(8;3), (18;18)}, {(5;10), (23;16)}

der beiden obigen Beispiele.



Im zweiten Beispiel sind es neunzehn Rechtecke.

Viele Wörter enden so, wie andere Wörter beginnen.

→ **Beispiel:**

Wettbewerb, Haftung, Ungemach, Erbschaft, machen

Wenn man diese Wörter, beginnend mit Wettbewerb, so aneinander reiht, dass die gemeinsamen Teile nur einmal auftreten, ergibt sich die Wörter-Kette „Wettbewerb-schaftungemachen“.

→ **Aufgabe**

Schreibe ein Programm, das aus einer Liste von Wörtern eine möglichst lange Wörter-Kette der oben beschriebenen Art erzeugt. Dabei soll das anzureihende Wort mit den drei letzten Buchstaben des vorigen Wortes beginnen. Die Wörter-Kette soll so lang sein, dass es in der Liste kein Wort mehr gibt, das mit den letzten drei Buchstaben der Wörter-Kette beginnt. Jedes Wort darf in der Wörter-Kette nur einmal vorkommen, so dass Endlos-Ketten, wie z. B. durch Verwendung des Wortes Kerker, nicht auftreten können.

Als Beispiel schicke uns eine von deinem Programm erzeugte möglichst lange Wörter-Kette aus den Wörtern der ersten Faust-Parts aus der ersten Studierzimmer-Szene aus Goethes Faust:

<http://www.bwinf.de/aufgaben/faust.txt>

Wettbewerbschaftungemachen

# Mitmachen – Schritt für Schritt

## → **Bearbeitung:**

Zu den Aufgaben erarbeite und sende uns jeweils Folgendes:

### → **Lösungsidee:**

Eine Beschreibung der Lösungsidee, unabhängig vom eingesandten Programm. Die Idee und die zu ihrer Beschreibung verwendeten Begriffe müssen aber im Programm nachvollziehbar sein.

### → **Programm-Dokumentation:**

Eine Beschreibung, wie die Lösungsidee im Programm und seinen Teilen realisiert wurde. Hinweise auf Besonderheiten und Nutzungsgrenzen, verlangte Eingabeformate usw.

### → **Programm-Ablaufprotokoll:**

Kommentierte Probeläufe des Programms, d. h. wann wird welche Eingabe erwartet, was passiert dann, welche Ausgabe erscheint usw. Mehrere unterschiedliche Beispiele, die die Lösung der Aufgabe und das Funktionieren des Programms verdeutlichen (teilweise in den Aufgabenstellungen vorgegeben). Bildschirm-Fotos sind zulässig.

### → **Programm-Text:**

Den kommentierten Quelltext des Programms in einer der gängigen höheren Programmiersprachen wie z. B. Pascal. Skriptsprachen sind zulässig, Maschinensprache nicht. Den Programmtext bitte ausdrucken, dabei aber auf nicht selbst geschriebene Teile (wie z. B. Funktionen der Entwicklungsumgebung und automatisch generierten Programmtext) verzichten.

### → **Programm:**

Das lauffähige Programm selbst auf einer CD oder 3,5"-Diskette. Dieser Datenträger muss auch den Programm-Text enthalten und unter Windows-Systemen lesbar sein. Ist kein Programm gefordert, strukturiere deine Bearbeitung der Aufgabenstellung entsprechend.

Bitte schicke deine Arbeit nach Aufgaben geordnet und geheftet auf einseitig bedrucktem DIN-A4-Papier. Nummeriere alle Blätter rechts oben und versieh sie mit deinem Namen. Die Texte sollen in Deutsch abgefasst sein. Verwende DIN-A4-Klarsichthüllen mit Lochrand (pro Aufgabe eine) oder loche die Blätter bitte. Beschrifte den Datenträger, der die Programme und Programm-Texte enthält, mit deinem Namen.

## → **Fragen zu den Aufgaben?**

per Telefon: 0228 / 37 86 46 jeweils von 17-19 Uhr am 8.9., 13.10. und 10.11.2004

per E-Mail: [bwinf@bwinf.de](mailto:bwinf@bwinf.de)

per Fax: 0228 / 37 29 000

per Brief: an den BWINF (siehe „→ Einsendung“)

Informationen (allgemeine Tipps und FAQ) gibt es auch im Internet unter [www.bwinf.de](http://www.bwinf.de). Meinungen und Fragen (aber keine Lösungsideen) zu den Aufgaben können auch in der BWINF-Newsgruppe [fido.ger.bwinf](mailto:fido.ger.bwinf) ausgetauscht werden.

## → **Anmeldung**

Die Anmeldung erfolgt, indem man seiner Einsendung ein ausgefülltes Anmeldeformular beilegt. Bei Gruppen muss jeder Teilnehmer ein Formular ausfüllen, Gruppenmitglieder ohne Anmeldeformular werden nicht gewertet!

Fülle das Anmeldeformular (Klappe des Aufgabenblattes oder eine Kopie davon) vollständig, korrekt und leserlich! aus. Insbesondere das Geburtsdatum muss angegeben sein, denn sonst kann die Einsendung nicht korrigiert werden. Teilnehmer, die die Schule bereits verlassen haben, geben bei „Klassenstufe“ bitte an, was sie zur Zeit machen. Erstteilnehmer nennen ggf. den Teilnehmer, der sie zum Mitmachen angeregt hat, mit Namen und Geburtsdatum.

Für die Anmeldung gibt es unter [www.bwinf.de](http://www.bwinf.de) auch ein Internet-Formular, das vor dem Einsendeschluss ausgefüllt werden kann. Bei dieser Online-Anmeldung wird eine Kennnummer vergeben; bei der Einsendung muss das Papierformular nur noch mit dieser Nummer, dem Namen und einer Unterschrift versehen werden. Wer sich per Internet anmeldet, erhält nach der Einsendung eine Eingangsbestätigung per E-Mail.

## → **Einsendung**

Einsendungen per E-Mail oder nur auf CD/Diskette ohne Ausdruck können wir leider nicht akzeptieren. Auch alle geforderten Beispiele müssen auf Papier dokumentiert sein. Es ist nicht auszuschließen, dass die Gutachter nur auf die Papierunterlagen zugreifen können.

### → **Sende alles an:** Bundeswettbewerb Informatik, Ahrstraße 45, 53175 Bonn

### → **Einsendeschluss ist der 15. November 2004** (Datum des Poststempels).

Verspätete Einsendungen können nicht berücksichtigt werden. Der Rechtsweg ist ausgeschlossen. Die Einsendungen werden nicht zurückgegeben. Der Veranstalter erhält das Recht, die Beiträge in geeigneter Form zu veröffentlichen. Wer wissen möchte, ob seine Einsendung angekommen ist, kann eine an sich selbst adressierte und frankierte Postkarte mitschicken oder die Online-Anmeldung nutzen.

## → **Bewertung**

Einsendungen werden danach bewertet,

- ob die Aufgaben vollständig und richtig bearbeitet wurden,
- ob die Ausarbeitungen gut strukturiert und verständlich sind und
- ob die (Programm-) Unterlagen vollständig, übersichtlich und lesbar sind.

# Anmeldung

23. Bundeswettbewerb Informatik 2004/2005, 1. Runde

Bitte in Druckschrift ausfüllen und Zutreffendes ankreuzen.

				
Geburtsdatum	Name, Vorname		<input type="checkbox"/>	<input type="checkbox"/>
Straße		männlich	weiblich	
Postleitzahl	Wohnort			
Telefon: Vorwahl/Durchwahl			E-Mail-Adresse	
Name der (ehemaligen) Schule			Schultyp	
Informatiklehrer/in		Klassenstufe (1-13)		
Straße der Schule				
Postleitzahl	Schulort	Bundesland der Schule		
Namen anderer Gruppenmitglieder				
Wieviele Stunden hast du gebraucht?				
Was hast du bei der Bearbeitung verwendet? Welche/s/n ... <input type="checkbox"/>				
	Programmiersprache?	Betriebssystem	Computer (typ)?	eigener
Hast Du an anderen Wettbewerben teilgenommen?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
		Informatik	Mathematik	sonstige
Wie hast du vom Wettbewerb erfahren?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	schon mal teilgenommen	Schule/Lehrer	ehemalige Teilnehmer	sonstiges
Geworben durch ehe- und diesmaligen Teilnehmer				
Name, Vorname, Geburtsdatum				
→ Diese Daten werden an niemanden weitergegeben, haben keinen Einfluss auf die Bewertung, aber dienen statistischen Zwecken. Sie werden ausschließlich für die Zwecke des Bundeswettbewerbs Informatik ausgewertet.				
→ Ich bin mit der Computerspeicherung dieser Daten einverstanden und versichere, dass ich die Aufgaben selbstständig bzw. mit den angegebenen Gruppenmitgliedern bearbeitet habe.				
Datum	Unterschrift			

# Beispielaufgabe

## Raff

Der berühmte, geldgierige Dagobert beobachtet den Geldmarkt in der Stadt Valuta im Viel-Länder-Eck immer sehr genau. Jeden Donnerstag kann man dort an vielen einzelnen Ständen Geld von einer Währung in eine andere tauschen. Dagobert hat festgestellt, dass er bei den unterschiedlichen Tageskursen oft durch mehrere geschickte Tauschvorgänge bei verschiedenen Händlern sein Geld vermehren kann. Dagobert geht über den Markt und notiert sich Ankaufskurse in einer Tabelle. Heute hat er folgende Tabelle zusammengestellt:

Währung	Kurs
1 Bärentaler	2,70 Mausmark
1 Entenpeseta	0,75 Krötendollar
1 Entenpeseta	0,70 Froschkronen
1 Entenpeseta	1,80 Wolfspfunde
1 Froschkrone	1,10 Krötendollar
1 Froschkrone	1,10 Entenpeseten
1 Krötendollar	2,00 Wolfspfunde
1 Krötendollar	1,90 Wolfspfunde
1 Krötendollar	1,05 Froschkronen
1 Mausmark	1,30 Entenpeseten
1 Mausmark	0,90 Krötendollar
1 Wolfspfund	0,70 Entenpeseten
1 Wolfspfund	0,20 Bärentaler
1 Wolfspfund	0,50 Mausmark

Er könnte z.B. pro Entenpeseta 0,75 Krötendollar bekommen, dann pro Krötendollar 2,0 Wolfspfunde und dann pro Wolfspfund 0,7 Entenpeseten. Mit solchem Tauschzyklus würde er aus je einer Entenpeseta  $(0,75 * 2,0 * 0,7) = 1,05$  Entenpeseten machen, also einen Gewinn von  $(5/3)\%$  pro Tausch erzielen. Dagobert sucht natürlich, von einer beliebigen Währung ausgehend, nach einem Tauschzyklus mit maximalem Gewinn pro Tausch.

### → Aufgabe

- Schreibe ein Programm, das die Tageskurstabelle einliest, prüft, ob es Tauschzyklen gibt, und gegebenenfalls einen mit maximalem Gewinn pro Tausch bestimmt.
- Dokumentiere die Programmresultate für drei verschiedene Tageskurstabellen. Eine soll die oben angegebene Tabelle sein.

### → Lösungsidee

Die Kurstabelle wird, ausgehend von allen enthaltenen Währungen, nach Zyklen durchsucht. Für alle gefundenen Zyklen wird der Gewinn pro Tausch bestimmt. Ein Zyklus wird ausgegeben, bei dem dieser Wert maximal ist.

Die Suche selbst wird rekursiv durchgeführt: Um einen Tauschweg von einer Währung A zu einer Währung B zu finden, wird für alle Währungen C, in die A getauscht werden kann, geprüft, ob es einen Tauschweg von C nach B gibt.

Allerdings wollen wir mehrere, verschiedene Tauschkurse zwischen zwei Währungen, wie von Krötendollar nach Wolfspfund, ausschließen. Da nach dem maximalen Gewinn pro Tausch gesucht ist, braucht nur der beste Kurs betrachtet werden. Außerdem schließen wir aus, dass in einem Tauschzyklus eine Währung mehrfach vorkommt. In diesem Fall gäbe es einen „inneren Zyklus“, den man prinzipiell beliebig oft durchlaufen könnte. Je nach Tausch-

kurs (z.B. 2,00) würde der Gewinn pro Tausch immer weiter wachsen, was nicht sinnvoll ist (würde man den Gewinn geometrisch mitteln, würden innere Zyklen gar nichts bringen).

### → Programm-Dokumentation

Wir haben die Lösungsidee in Python implementiert. Das Programm besteht aus drei Prozeduren und dem Hauptprogramm. Das Hauptprogramm liest zuerst eine Kurstabelle aus einer Datei ein (Prozedur **liesKursTabelle**). Anschließend werden für jede Währung die Tauschwege zu sich selbst bestimmt (Prozedur **findeWege**). Das Ergebnis ist eine Liste von Zyklen (die selbst wiederum Listen von Währungsnamen sind, bei denen das erste Element gleich dem letzten ist). Dann wird für jeden Zyklus berechnet, mit welchem Faktor sich die jeweilige Währung durch den Zyklus vermehrt hat (Prozedur **vermehrungsfaktor**). Zuletzt werden diese Faktoren gemittelt und dann bestimmt, an welcher Stelle der Zyklenliste ein Zyklus mit maximalem Wert zu finden ist. Dieser Zyklus wird mit Vermehrungsfaktor und Gewinn pro Tausch ausgegeben.

Zu den wichtigen Prozeduren im einzelnen: Die Prozedur **liesKursTabelle** verwendet das Python-Modul **csv** zum Einlesen einer Datei, die Zeilen der Form **Währung1, Währung2, Wechselkurs** enthält (Beispiel: **BT,MM, 2.70**), also „comma-separated values“. Die Währungsnamen werden abgekürzt, so dass es kein Problem mit unterschiedlichen Formen gibt. Die Kurse werden in einem zweidimensionalen Feld gespeichert.

Die Prozedur **findeWege** durchsucht dieses Feld und führt den aktuell betrachteten Weg im letzten Parameter mit. Ist die aktuelle Währung gleich dem Ziel, kann dieser Weg abgespeichert werden. Ansonsten wird **findeWege** rekursiv aufgerufen, und zwar mit allen möglichen Tauschzielen der aktuellen Währung als neuem Ausgangspunkt.

### → Programm-Ablaufprotokolle

Aus Platzgründen kann hier nur ein Ablaufprotokoll abgedruckt werden. Die Kurstabelle aus der Aufgabenstellung befindet sich in der Datei **tabelle1.txt**, in der oben angegebenen Form. Damit ergibt sich das Ablaufprotokoll:

```
> python raff.py tabelle1.txt
Lade tabelle1.txt ...
Suche Zyklen...
Berechne Ergebnisse der Zyklen...
Der beste Tauschzyklus ist: ['WP', 'EP', 'WP']
Aus einer Währungseinheit werden in diesem Zyklus
1.26 Einheiten.
Das bedeutet einen Gewinn von 13.0 Prozent pro Tausch.
>
```

### → Programm-Text

```
#!/usr/bin/python
# -*- coding: iso-8859-15 -*-

# raff.py - Lösung für die Aufgabe "Raff" des 22. BWINF
# nach Manuel Holtgrewe

def liesKursTabelle(filename):
    """ Diese Funktion liest eine Kurstabelle aus der Datei
    mit dem Namen "filename" und gibt sie als Feld zurück. """

    tabelle = {}
    # Hilfsmodul für die Bearbeitung von CSV-Dateien:
    import csv
    reader = csv.reader(file(filename))
    for zeile in reader:
        w1, w2, kurs = zeile[0], zeile[1], float(zeile[2])
        if not tabelle.has_key(w1):
            tabelle[w1] = {}
        if not tabelle[w1].has_key(w2):
            tabelle[w1][w2] = kurs
    # Tabelle nur mit besserem Kurs aktualisieren
    elif kurs > tabelle[w1][w2]:
        tabelle[w1][w2] = kurs
    return tabelle

def findeWege(tabelle, waehrung, zielw, wege, weg):
    """ Diese Funktion durchsucht die "tabelle" nach einem Tauschweg
    von "waehrung" aus nach der Währung "zielw"
    und gibt gefundene Wege im Parameter "wege" zurück """

    # Pfad um aktuellen Knoten verlängern...
    weg = weg + [waehrung]
    # Haben wir damit einen gültigen Weg gefunden?
    if (waehrung == zielw) and (len(weg) > 1):
        wege.append(weg)
        return
    # Prüfen, ob es von hier aus weiter geht...
    if not tabelle.has_key(waehrung):
        return
    # Suche auf allen Kanten nach möglichen Pfaden
    for w in tabelle[waehrung]:
        if (w not in weg) or (w == zielw):
            findeWege(tabelle, w, zielw, wege, weg)

def vermehrungsfaktor(tabelle, zyklus):
    """ Diese Funktion berechnet, um welchen Faktor das Geld sich beim
    Umtauschen auf diesem Zyklus vermehrt/verringert. """
    ergebnis = 1
    for i in range(1, len(zyklus)):
        ergebnis = ergebnis * tabelle[zyklus[i-1]][zyklus[i]]
    return ergebnis

### Hauptprogramm ###

# Hole den Dateinamen der Kurstabelle aus der Kommandozeile
import sys
if len(sys.argv) != 2:
    print "Fehler beim Aufruf. Bitte so aufrufen:"
    print " %> python raff.py kursdatei"
    raise SystemExit()
datei = sys.argv[1]

# Lade die Kurstabelle
print "Lade", datei, "..."
tabelle = liesKursTabelle(datei)

print "Suche Zyklen..."
zyklen = []
# Zyklen sind Wege von einer Währung zu sich selbst
for waehrung in tabelle:
    findeWege(tabelle, waehrung, waehrung, zyklen, [])

# Berechne Faktoren und Gewinn pro Tausch
print "Berechne Ergebnisse der Zyklen..."
faktoren = []
for i, zyklus in enumerate(zyklen):
    faktor = vermehrungsfaktor(tabelle, zyklus[i])
    faktoren.append(faktor)
gemittelte_faktoren = []
for i, zyklus in enumerate(zyklen):
    mittel = (faktoren[i] - 1) * 100 / (len(zyklus[i]) - 1)
    gemittelte_faktoren.append(mittel)

# Suchen des besten Faktors
max_index = gemittelte_faktoren.index(max(gemittelte_faktoren))

print "Der beste Tauschzyklus ist:", zyklen[max_index]
print "Aus einer Währungseinheit werden in diesem Zyklus"
print faktoren[max_index], "Einheiten."
print "Das bedeutet einen Gewinn von", gemittelte_faktoren[max_index], "%"
print "Prozent pro Tausch."
```