

Fähre füllen

Von Oberdellendorf verkehrt eine Autofähre über die Wesu nach Niederdellendorf. Die nächste Brücke ist weit entfernt; deshalb ist die Fähre sehr beliebt, und an der Anlegestelle ist die Warteschlange lang. Da ist es wichtig, so viele Autos wie möglich auf der Fähre unterzubringen.

Der Fährbegleiter kann die Fahrzeuge – Pkws, Pkws mit Anhängern, Kleinlaster usw. – bei der Aufahrt jeweils in eine der drei, jeweils 20 m langen Parkbahnen der Fähre dirigieren. Er weist wiederholt das Fahrzeug am Kopf der Warteschlange einer Parkbahn zu und kennt dabei die Länge dieses Fahrzeuges, aber nicht die der nachfolgenden Fahrzeuge. Zwischen zwei aufeinanderfolgenden Fahrzeugen auf einer Parkbahn wird ein Abstand von 30 cm frei gelassen. Sobald das Fahrzeug am Kopf der Warteschlange nicht mehr untergebracht werden kann, legt die Fähre ab.

Junioraufgabe 1

1. Finde zwei einfache Strategien A und B, die der Fährbegleiter bei der Zuweisung der Fahrzeuge zu den Parkbahnen anwenden könnte. Wähle die Strategien so, dass bei manchen Folgen von wartenden Fahrzeugen die Strategie A und bei manchen anderen die Strategie B mehr Fahrzeuge auf der Fähre unterbringt.

2. Schreibe ein Programm, das eine Folge von Längen von wartenden Fahrzeugen einliest und überschichtlich ausgibt, wie die Strategien A und B mit dieser Eingabe die Fähre beladen.

3. Wende dein Programm auf die folgenden drei Beispiele an. Angegeben sind jeweils die Längen einiger Fahrzeuge (in Metern) in der Reihenfolge, in der sie in der Warteschlange stehen:

6,96; 5,06; 3,77; 3,95; 3,91; 3,54; 4,26; 4,03;
5,43; 4,04; 4,43; 4,12; 2,78

4,14; 3,63; 3,92; 7,95; 5,23; 3,30; 4,86; 15,06

5,23; 4,41; 3,33; 13,13; 9,12; 4,38; 6,34; 5,37;
4,11; 3,74; 10,62

Belege mit diesen oder anderen Beispielen, dass manchmal Strategie A und manchmal Strategie B besser abschneidet.

Zahlenspiel

Ulla gibt regelmäßig Nachhilfe. Immer wieder wollen die Schüler und Schülerinnen das Kürzen von Brüchen üben. Deshalb möchte Ulla gute Übungsaufgaben zum Kürzen automatisch erzeugen lassen: Ein Programm soll Brüche a/b wie $144/56$ erzeugen, die dann von den Schülern zu p/q möglichst weit gekürzt werden, also $144/56 = 18/7$.

Sie benötigt Aufgaben von drei Schwierigkeitsstufen: leicht, mittel und schwer. Die Schwierigkeit der Aufgabe „Kürze a/b “ (mit Ergebnis p/q) hängt von der Größe der Zahlen a , b , p und q ab.

Ulla nennt die Gesamtanzahl der Ziffern in a und b die *Länge* des Bruchs a/b ; z.B. hat $144/56$ die Länge 5. Sie legt die Schwierigkeitsstufen wie folgt fest:

Stufe	Länge von a/b	Bedingung für p/q
leicht	4	$p+q \leq 10$
mittel	5	$10 < p+q \leq 20$
schwer	5	$20 < p+q \leq 30$

Die für eine Stufe angegebenen Bedingungen müssen beide erfüllt sein. Außerdem müssen die Aufgaben weitere Bedingungen erfüllen:

- > a/b soll tatsächlich gekürzt werden können, d.h. es soll $q < b$ sein.
- > Es soll immer $a \neq b$ sein.

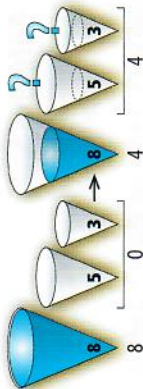
Wie man sieht, ist die Aufgabe „Kürze $144/56$ “ schwer, und „Kürze $22/77$ “ ist leicht.

Junioraufgabe 2

Hilf Ulla und schreibe ein Programm, das für eine gegebene Schwierigkeitsstufe eine gewünschte Anzahl von zufällig erzeugten Aufgaben der angegebenen Stufe mit ihren Lösungen ausgibt.

Faires Füllen

In einem Jahrhunderte alten Rätsel treffen sich zwei Personen. Die erste hat einen vollen Behälter mit acht Maßen Wein bei sich, die andere zwei leere Behälter mit einem Fassungsvermögen von drei und fünf Maßen. Können sie den Wein ohne Zuhilfenahme von anderen Behältern so umfüllen, dass jeder vier Maße in seinen Behältern hat? Ja, aber ganz einfach findet man die Lösung nicht.



Wir wollen dieses Rätsel hier verallgemeinern und mit Computerhilfe lösen lassen. Wieder treffen sich zwei Personen. Jede trägt eine bestimmte Menge von Behältern mit sich. Es ist bekannt, wie groß und wie voll diese Behälter zunächst sind.

Aufgabe 1

Schreibe ein Programm, das diese Daten einliest und dann feststellt, ob ein Umfüllen so möglich ist, dass beide Personen die gleiche Gesamtmenge

Weins in ihren Behältern vorfinden. Wenn ja, dann soll das Programm einen übersichtlichen Plan ausgeben, wie das Umfüllen in einer minimalen Anzahl von Schritten vollführt werden kann. Ein Schritt ist dabei das Gießen des Weins von einem Behälter A in einen Behälter B, bis entweder A leer oder B voll ist.

Wende dein Programm auf das ursprüngliche Rätsel an sowie auf die Beispiele, die du unter www.bundeswettbewerb-informatik.de findest.

Mobile

Ein Mobile besteht aus waagerechten Balken, senkrechten Drähten und an Drähten hängenden Figuren. Die Gewichte der Balken und Drähte können vernachlässigt werden, während die Gewichte der Figuren die möglichen Formen des Mobiles beeinflussen.

Jeder Balken hat einen Aufhängepunkt und weitere Stellen, an denen tiefer hängende Balken oder Figuren an ihm befestigt sind. Der Balken ist balanciert, wenn die Gewichte der an ihm befestigten Strukturen, multipliziert mit den jeweiligen Abständen ihrer Befestigung vom Aufhängepunkt des Balkens, die Summe Null haben, falls Abstände auf einer Seite vom Aufhängepunkt mit 1, die auf der anderen Seite mit -1 multipliziert werden. Das gesamte Mobile ist balanciert, wenn alle seine Balken balanciert sind.

Zum Beispiel ist das unten abgebildete Mobile balanciert, denn:

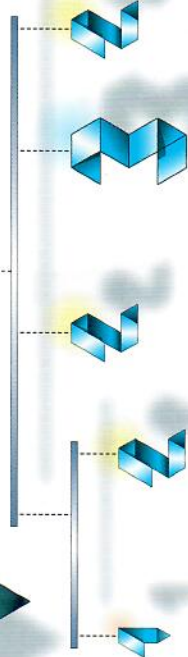
$$1 \cdot 2 + 2 \cdot (-1) = 0 \text{ und } (1+2) \cdot 4 + 2 \cdot 1 + 3 \cdot (-2) + 2 \cdot (-4) = 0.$$

Aufgabe 2

1. Erfinde und beschreibe eine einfache textuelle Darstellung von Mobiles, die eine Überprüfung der Balanciertheit erlaubt. Nicht für diesen Zweck relevante Aspekte, wie zum Beispiel die Länge der Drähte, sollen außer Acht gelassen werden. Zeige unter anderem das unten abgebildete Mobile in deiner Darstellung.

2. Schreibe ein Programm, das eine Folge $w_1, \dots, w_n$ von Gewichten von n Figuren einliest und die textuelle Darstellung eines balancierten Mobiles mit den n Figuren ausgibt. Aus ästhetischen Gründen sollen an keinem Balken mehr als vier Strukturen befestigt sein.

Freiwillige Zusatzaufgabe, ohne Einfluss auf die Bewertung: Erweitere dein Programm so, dass es auch eine nette Zeichnung des gefundenen Mobiles produziert.



Buffet-Lotterie

Bei der Endrunde des Bundeswettbewerbs Informatik haben die Teilnehmer es satt, Warteschlangenunter dem großen Buffet im engen Korridor und Opfer der Last-Come-Longest-Hungry-Mentalität zu sein. Stattdessen soll ganz elegant und zivilisiert ausgelost werden, wer als Nächster das Buffet aufsuchen darf.

Die 28 Teilnehmer stellen sich dazu in einem großen Kreis auf und sagen den Satz

In-for-ma-tik kann uns wei-sen,
wer als Näch-ster kommt zum Spei-sen

wiederholt laut auf. Wie beim Ene-Mene-Muh spricht jeder Teilnehmer nur eine Silbe, dann ist sein rechter Nachbar an der Reihe. Und wer die letzte Silbe des Satzes sagt, ist der Glückliche, der den Kreis verlässt und als Nächster seinen Hunger stillen darf.

Eine Teilnehmerin hat aber Geburtstag. Sie spricht natürlich die allererste Silbe, und als besondere Gunst darf sie, wann immer sie an der Reihe ist, statt einer auch zwei Silben sprechen – wenn sie das denn will. Da ihr Magen knurrt, möchte sie ihren Vorteil dazu nutzen, so schnell wie möglich zum Buffet zu kommen.

Aufgabe 3

Schreibe ein Programm, das für eine gegebene Anzahl von Teilnehmern berechnet, wann das Geburtstagskind zwei Silben sprechen soll, um sich den bestmöglichen Platz in der Buffetreihenfolge zu verschaffen.

Dokumentiere die Wirkungsweise deines Programms für verschiedene Teilnehmerzahlen, unter anderem für die oben genannten 28 Teilnehmer.