AUFGABENBLATT ZUM

20. BUNDESWETTBEWERB INFORMATIK

■ 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 von der Bundesministerin für Bildung und Forschung. Er steht unter der Schirmherrschaft des Bundespräsidenten. Die Träger des Wettbewerbs sind die GI und die GMD – Forschungszentrum Informationstechnik GmbH. Die Gestaltung des Wettbewerbs und die Auswahl der Sieger obliegen dem Beirat; Vorsitzender: Prof. Dr. Ingo Wegener, Universität Dortmund. Die Auswahl und Entwicklung von Aufgaben und die Festlegung von Bewertungsverfahren übernimmt der Aufgabenausschuss; Vorsitzende: StD Vera Reineke, Hannover. 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 ohne größere Informatikkenntnisse 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 12.11.1979 geboren wurden. Sie dürfen jedoch zum 1.9.2001 noch nicht ihre (informatikbezogene) Ausbildung abgeschlossen oder eine Berufstätigkeit aufgenommen haben. Ebenfalls ausgeschlossen sind Personen, die zum Wintersemester 2001/2002 oder früher ihr Studium an einer Hochschule/Fachhochschule aufnehmen bzw. aufgenommen haben. Jugendliche, die nicht deutsche Staatsangehörige sind, müssen wenigstens vom 1.9. bis 12.11.2001 ihren Wohnsitz in Deutschland haben oder eine staatlich anerkannte deutsche Schule im Ausland besuchen.

■ ALS ANERKENNUNG …

Jeder Teilnehmer der ersten Runde erhält eine Urkunde für die Teilnahme bzw. die erfolgreiche Teilnahme bei richtiger Lösung von wenigstens drei der fünf Aufgaben. Erfolgreiche Teilnahme berechtigt zum Eintritt in die zweite Runde. Die Besten der zweiten Runde werden zur dritten Runde eingeladen, die im Herbst 2002 von der Dresdner Bank 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 Bundessieg, aber auch für andere besondere Leistungen Geld- und Sachpreise vorgesehen, u.a. ein Aufenthalt an einer Sommerschule in den USA sowie Einladungen zu Schülerakademien.

■ … TEILNAHME AN DER INFORMATIK-OLYMPIADE

Aus dem Teilnehmerkreis der Endrunde werden die Kandidaten festgelegt, aus denen im Laufe mehrerer Trainingsrunden das vierköpfige deutsche Team ausgewählt wird, das an der Internationalen Olympiade in Informatik 2003 in den USA teilnimmt.

■ SONDERPREISE BEIM 20. BWINF – TEILNEHMER WERBEN TEILNEHMER

Der Bundeswettbewerb Informatik wird in diesem Jahr zum 20. Mal ausgeschrieben. Dieses kleine Jubiläum soll gefeiert werden, und deshalb gibt es beim 20. BWINF einiges zu gewinnen. Insgesamt 60 aktuelle Versionen von SuSE Linux werden verlost, und zwar u.a. unter den besten Erstrundenteilnehmern sowie unter Lehrkräften und Schulen mit besonders vielen BWINF-Teilnehmern. In der zweiten Runde wird für herausragende Leistungen ein Apple iMac vergeben. Auch der Sonderpreis Mensch-Computer-Interaktion (MCI) wird wieder gemeinsam mit dem gleichnamigen GI-Fachbereich ausgeschrieben, diesmal in der zweiten Runde.
Zu gewinnen ist die Spracherkennungssoftware ViaVoice von IBM.

Unter dem Motto Teilnehmer werben Teilnehmer wollen wir den Austausch zwischen erfahrenen und neuen BWINF-Teilnehmern fördern. An der Verlosung von 20 SuSE-Linux-Paketen können Personen teilnehmen, die schon am BWINF teilgenommen haben, beim 20. BWINF wieder dabei sind und eine Person ohne BWINF-Erfahrung für die Teilnahme am 20. BWINF werben konnten. Der oder die Geworbene muss dazu Namen und Geburtsdatum des/der Werbenden bei der Anmeldung angeben.

Auf dem unten abgebildeten sechseckigen Spielbrett soll ein Spielstein so viele Felder in gerader Linie weiterhüpfen, wie die Zahl auf dem Feld, auf dem er steht, angibt. Begonnen wird auf dem Feld in der Mitte.
Der Anfangszug darf in eine beliebige der sechs möglichen Richtungen erfolgen. Bei jedem weiteren Zug darf jeweils nur in derselben Richtung weitergehüpft werden – oder in eine rechts oder links benachbarte Richtung. Die Hüpferei ist erfolgreich zu Ende, wenn das Ausgangsfeld wieder erreicht wird.

Richtungen:

Erlaubt:

Aufgabe

Schreibe ein Programm, welches einen erfolgreichen Hüpfweg sucht und ein Protokoll des Weges ausgibt (pro Hüpfer: die Richtung und Hüpfweite).
Falls das Ausgangsfeld nicht erreicht werden kann, soll dies an Stelle des Protokolls ausgegeben werden.

Lösungsidee

Der erste Schritt zur Lösung der Aufgabe besteht darin, sich eine geeignete Datenstruktur für das sechseckige Spielbrett zu überlegen. Das Brett lässt sich in eine quadratische Matrix schreiben, deren Größe sich an der „Mittelzeile" orientiert, die also 9x9 Einträge hat. Der obere Teil des Spielbrettes wird dann am linken Rand der Matrix orientiert, der untere Teil am rechten Rand. Alle verbleibenden Stellen der Matrix können z. B. mit 0 aufgefüllt werden. Das sieht dann so aus:

2	4	1	5	4	0	0	0	0
1	4	5	2	3	2	0	0	0
2	1	3	3	2	5	2	0	0
2	1	3	2	1	2	2	4	0
4	3	4	3	3	4	2	3	2
0	2	2	1	3	2	2	1	2
0	0	2	2	2	3	4	3	2
0	0	0	1	3	1	2	1	4
0	0	0	0	2	1	6	3	2

Für jede Richtung wird festgelegt, wie eine Bewegung auf dem Brett durch die Wahl eines neuen Matrix-Elements simuliert wird.
Zum Finden eines Weges wird das Spielbrett systematisch durchsucht, und zwar rekursiv. Dabei wird jedoch eine Einschränkung beachtet: Ein Feld darf in einem Hüpfweg nicht zweimal von der gleichen Richtung kommend betreten werden. Damit werden möglicherweise endlose Rundläufe vermieden.

Programm-Dokumentation

Die oben beschriebene Matrix für das Spielbrett wird in dem 9x9 großen Integer-Array `brett` gespeichert.
Die Richtungen werden mit Hilfe des Enumerationstyps `richtungen` mit Zahlen von 0 bis 5 identifiziert. Die Richtungsnamen sind im Array `bezeichner` passend abgelegt.
Das Array `bewegung` legt für jede Richtung fest, wie sich die Matrixkoordinaten bei einem Schritt in diese Richtung verändern; bei mehreren Schritten muss multipliziert werden. Im Array `weg` werden die Koordinaten aller im aktuellen Weg besuchten Felder für die Ausgabe gespeichert.
Das Integer-Array `besucht` ist genau so groß wie das Array `brett`. Es dient dazu, für jedes Feld markieren zu können, in welcher Richtung es beim aktuellen Weg schon besucht wurde. Für jede Richtung wird dabei ein eigenes Bit benutzt. Das Bit für eine Richtung z kann mit einer XOR-Zuweisung und einer Shift-Operation (^= 1 << z) gesetzt und wieder gelöscht werden.
Die Funktion `suche` gibt einen gefundenen Weg aus, wenn das Ausgangsfeld wieder erreicht ist. Ansonsten wird für alle drei möglichen Fortsetzungsrichtungen (die im Kopf der for-Schleife mit Hilfe des Modulo-Operators % berechnet werden) das nächste Feld bestimmt und von dort aus weitergesucht. Dabei wird die Rekursionstiefe als Parameter mitgeführt, da sie gleichzeitig angibt, an wievielter Stelle eines Weges das aktuelle Feld besucht wurde.
Das Hauptprogramm ruft die Funktion `suche` mit den einander gegenüberliegenden Richtungen O und W auf, um alle sechs vom Startfeld aus möglichen Richtungen abzuarbeiten.

Programm-Ablaufprotokoll

Das Programm erwartet keine Eingaben, das Spielfeld ist ja festgelegt. Nach weniger als einer Sekunde werden 46 gefundene Lösungen ausgegeben, darunter die folgende mit der kürzesten Weglänge:

```
18 Züge: 3 SO, 1 SW, 3 W, 2 NW, 2 NW, 4 NO, 2
O, 1 SO, 2 SW, 2 SO, 2 O, 1 NO, 3 NW, 3 W, 4
SW, 2 SO, 1 O, 3 NO
```

Programm-Text

```c
#include <stdio.h>
#include <string.h>

int brett[9][9] = {  // Werte des Spielbretts
  2, 4, 1, 5, 4, 0, 0, 0, 0,
  1, 4, 5, 2, 3, 2, 0, 0, 0,
  2, 1, 3, 3, 2, 5, 2, 0, 0,
  2, 1, 3, 2, 1, 2, 2, 4, 0,
  4, 3, 4, 3, 3, 4, 2, 3, 2,
  0, 2, 2, 1, 3, 2, 2, 1, 2,
  0, 0, 2, 2, 2, 3, 4, 3, 2,
  0, 0, 0, 1, 3, 1, 2, 1, 4,
  0, 0, 0, 0, 2, 1, 6, 3, 2,
};

enum richtungen { O, SO, SW, W, NW, NO };
char* bezeichner[6] = { "O", "SO", "SW", "W", "NW", "NO" };

int bewegung[6][2] =  // dx und dy der sechs Bewegungsrichtungen
  { { 1, 0 }, { 1, 1 }, { 0, 1 }, { -1, 0 }, { -1, -1 }, { 0, -1
} };

int weg[100][2];     // speichert den gegangenen Weg
int besucht[9][9];   // benötigt zur Markierung besuchter Felder

void suche(int x, int y, int ausrichtung, int tiefe)
{
  int z;
  int newx, newy;

  if (x == 4 && y == 4 && tiefe != 0)  // Ausgangsposition wieder
  erreicht
  {
    printf("%d Züge: ", tiefe);
    for (z = 0; z < tiefe; z++)
    // Ausgabe des gegangenen Weges
      printf("%d %s%s", weg[z][1], bezeichner[weg[z][0]],
       (z < tiefe - 1 ? ", " : "\n"));
    return;
  }
  // Fortsetzung der Suche in alle drei möglichen Richtungen:
  for (z = (ausrichtung + 5) % 6;
       z != (ausrichtung + 2) % 6;
       z = (z + 1) % 6)
  {
   weg[tiefe][0] = z;            // Speichern von Ausrichtung und
   weg[tiefe][1] = brett[y][x]; // Sprungweite für spätere Ausgabe

    newx = x + brett[y][x] * bewegung[z][0];  // neue x-Koordinate
    newy = y + brett[y][x] * bewegung[z][1];  // neue y-Koordinate

    // Prüfung, ob die neue Position innerhalb des Bretts liegt
    // und noch nicht in gleicher Richtung begangen worden ist:

    if (newx >= 0 && newy >= 0 && newx <= 8 && newy <= 8 &&
      brett[newy][newx] && !(besucht[newy][newx] & 1 << z))
    {
      besucht[newy][newx] ^= 1 << z;      // Markierung setzen
      suche(newx, newy, z, tiefe + 1);    // rekursiver Aufruf
      besucht[newy][newx] ^= 1 << z ;     // Markierung entfernen
    }
  }
}

int main()
{
  memset(besucht, 0, sizeof(besucht));
  suche(4, 4, O, 0);  // Aufruf für die Startrichtungen NO, O, SO
  suche(4, 4, W, 0);  // und zweiter Aufruf für NW, W, SW
  return 0;
}
```

Lösung nach Alexander Hullmann und Alexander Kreuzer

Design: Kuhn, Kammann & Kuhn, Köln Illustration: Chris Janata, Köln Druck: DFS Druck und Verlag, Köln

■ GRUSSWORT DER BUNDESMINISTERIN FÜR BILDUNG UND FORSCHUNG

Informationstechnologien (IT) schaffen Wohlstand und Arbeitsplätze. Keine andere Technologie hat in so kurzer Zeit so viele gesellschaftliche Bereiche durchdrungen und zu fundamentalen Veränderungen von Arbeitswelt und Privatleben geführt. Internet, E-Mail und Handy sind bereits Alltag für Jugendliche geworden. Sie sind es auch, die die weitere Entwicklung der Informations- und Wissensgesellschaft prägen werden. Wer IT täglich nutzt, wird andere Visionen von der IT-Zukunft haben, die sich deutlich von denen Älterer unterscheiden. Das gilt es zu nutzen. Dazu sind aber gute Kenntnisse der Informatik notwendig.

Wir brauchen IT-Fachleute, die die Zukunft unseres Landes mitgestalten. Immer noch gibt es viel zu wenig IT-Expertinnen und -Experten. Die Bundesregierung hat mit der „Offensive zum Abbau des IT-Fachkräftemangels", dem „Sofortprogramm der Bundesregierung und IuK-Wirtschaft zur Deckung des IT-Fachkräftebedarfs in Deutschland" und zahlreichen anderen Maßnahmen im Rahmen des Handlungskonzepts „Anschluss statt Ausschluss – IT in der Bildung" die Voraussetzungen geschaffen, dass künftig genügend IT-Personal zur Verfügung steht.

Bei der Zahl der Lehrstellen in den neuen IT- und Medienberufen sind allein im Jahre 2000 über 25.000 neue Verträge abgeschlossen worden. Das vereinbarte Ziel im Bereich der Erstausbildung wurde damit deutlich früher erreicht und mit über 50.000 Lehrstellen sogar um 10.000 überschritten. Wenn die Wirtschaft auch künftig so zahlreich ausbildet, werden bis zum Jahr 2003 die im Sofortprogramm zur Green-Card-Initiative vereinbarten 60.000 Ausbildungsplätze nachhaltig verwirklicht werden können. Ebenso hat das „Sofortprogramm zur Weiterentwicklung des Informatikstudiums – WIS" etwas zur Verbesserung der Situation im Hochschulbereich getan. Dafür wurden zusammen mit den Ländern 100 Millionen Mark zur Verfügung gestellt. Die Zahl der Studierenden ist dort auch wieder kräftig gestiegen.

Kreativität und Leistungsbereitschaft der kommenden IT-Elite stehen im Vordergrund des Bundeswettbewerbs Informatik. Wie kein Zweiter eignet er sich dazu, IT-Begeisterung und IT-Begabung zu fördern. Ich wünsche allen Teilnehmerinnen und Teilnehmern zündende Ideen und viel Erfolg. Insbesondere möchte ich aber die weiblichen IT-Fans ermuntern, an dem Wettbewerb zahlreich teil zu nehmen und der Männerdomäne IT den notwendigen weiblichen Touch zu geben.

Edelgard Bulmahn

■ GESELLSCHAFT FÜR INFORMATIK E.V.

Die Gesellschaft für Informatik e.V. (GI) ist mit rund 22.000 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 200 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.
Weitere Info: **www.gi-ev.de**

■ FRAUNHOFER-GESELLSCHAFT, IuK-GRUPPE

Die Fraunhofer-Gesellschaft zur Förderung der Angewandten Wissenschaften e.V. führt mit breitem thematischen Spektrum anwendungsorientierte Forschung und Entwicklung durch. Sie kooperiert dabei mit nationalen und internationalen Partnern aus Wirtschaft und Wissenschaft. Mit 56 Instituten und mehr als 11.000 Mitarbeiterinnen und Mitarbeitern ist die Fraunhofer-Gesellschaft eine der größten Forschungsinstitutionen in Europa.

Die Unternehmensgruppe Informations- und Kommunikationstechnik bündelt die Forschungsarbeit von 15 Fraunhofer-Instituten in der Informations-, Kommunikations- und Medientechnik. Rund 2000 Wissenschaftlerinnen und Wissenschaftler entwickeln hier innovative IT-Lösungen für Wirtschaftsunternehmen; in Projekten der Vorlaufforschung entstehen die IT-Technologien von morgen.

bmb+f GI Fraunhofer IuK Gruppe Informations- und Kommunikationstechnik

Unter der Schirmherrschaft des Bundespräsidenten

Träger des Wettbewerbs:
Bundesministerium für Bildung und Forschung,
Gesellschaft für Informatik e.V. (GI),
Gruppe Informations- und Kommunikationstechnik der Fraunhofer-Gesellschaft

■ ALLGEMEINE HINWEISE:

Zu den Aufgaben 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 pas siert 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. Maschinensprache ist nicht zulässig. Den Programmtext bitte ausdrucken, dabei aber auf nicht selbst geschriebene Teile (z. B. verwendete Funktionen der Entwicklungsumgebung, automatisch generierter Programmtext) verzichten.

Programm:
Das lauffähige Programm selbst auf einer 3,5"-Diskette oder CD, die unter DOS bzw. Windows95 gelesen werden kann. Die Diskette/CD muss auch den Programm-Text enthalten.

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 übersichtlich und lesbar sind.

Bitte schicke deine Arbeit nach Aufgaben geordnet und geheftet auf einseitig bedrucktem DIN-A4-Papier. Endlospapier schneide bitte entsprechend zu. Nummeriere alle Blätter rechts oben und versieh sie mit deinem Namen. Die Texte sollen in Deutsch abgefasst sein. Falls du einige DIN-A4-Klarsichthüllen mit Lochrand zur Hand hast, stecke bitte jeweils alles, was zu einer Aufgabe gehört, in eine solche Hülle. Andernfalls loche die Blätter bitte.

Beschrifte die Diskette oder CD, die die Programme und Programm-Texte enthält, mit deinem Namen.

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.
Bei Gruppen muss **jeder** Teilnehmer ein Formular ausfüllen, Gruppenmitglieder ohne Anmeldeformular werden nicht gewertet!
Für die Anmeldung gibt es unter **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.
Einsendungen per E-mail oder nur auf Diskette/CD ohne Ausdruck können wir leider nicht akzeptieren. Auch alle geforderten Beispiele müssen auf Papier dokumentiert sein.

Sende alles an:
Bundeswettbewerb Informatik
Ahrstraße 45
53175 Bonn

Für Fragen zu den Aufgaben gibt es eine Hotline:
Telefonisch unter 0228 / 37 86 46 jeweils von 17-19 Uhr am 21.9., 22.10. und 9.11.
oder E-mail an: bwinf@bwinf.de
oder schriftlich an die obige Adresse bzw. per Fax an 0228 / 37 29 000.
Informationen (FAQ) gibt es auch im Internet unter www.bwinf.de
Meinungen und Fragen (aber keine Lösungsideen) zu den Aufgaben können auch in der BWINF-Newsgroup fido.ger.bwinf ausgetauscht werden.

Einsendeschluss ist der 12. November 2001 (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 sich auf den BWINF-Webseiten mit E-mail-Adresse anmelden.

■ AUFGABE 1: RIBO-NATTER

Die Ribo-Natter (Kosename: RNA) ist eine lange Schlange mit vielen farbigen Ringeln, die in folgenden Farben vorkommen:
Aquamarin, **U**mbra, **G**rasgrün und **C**yan

Diese Schlangenart ist ganz besonders eitel und möchte dem Betrachter ein möglichst gefälliges Aussehen präsentieren. Daher legt sie sich ganz flach so auf den Boden, dass – ihrer Meinung nach – eine farblich besonders harmonische Schlängelung entsteht. Dabei gibt es vier Punkte zu beachten:

1) Sie möchte dem Betrachter gerne viele Harmoniestellen zeigen. Das sind Stellen, an denen **A** mit **U** bzw. **C** mit **G** in Berührung kommt.

2) Ein Ringel darf zu höchstens einer Harmoniestelle gehören.

3) Damit die Schlängelung nicht zu geknickt aussieht, müssen zwischen zwei Ringeln, die miteinander eine Harmoniestelle bilden, mindestens drei Ringel sein, die zu keiner Harmoniestelle gehören.

4) Besonders schön sind Harmonieabschnitte, d. h. mehrere direkt aufeinander folgende Harmoniestellen.

Für diese Aufgabe nehmen wir an, dass die Ringel einer Schlange fortlaufend nummeriert sind, und zwar beginnend mit 1 vom Kopf her. Die oben abgebildete Schlange kann also wie folgt beschrieben werden:

Ringel:	U	A	U	G	C	G	A	A	A	G	U	C	U	G	U	G	A	U	C	C	A	G	A	U	C	G	C	A	C
Nr.:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

Eine Harmoniestelle entspricht dann einem Nummernpaar, z. B. (3,28); ein Harmonieabschnitt kann durch ein Paar von Nummernbereichen beschrieben werden: (3-7,28-24).

Eine komplette Schlängelung wird durch die entstandenen Harmonieabschnitte beschrieben; die Schlängelung aus der Abbildung lautet also (3-7,28-24) (12-14,22-20).

» Aufgabe

1. Schreibe ein Programm, das eine Schlange als Folge von Farbringeln einliest und ihren längsten Harmonieabschnitt findet. Demonstriere das Programm an drei Beispielen.
Eines dieser Beispiele soll die folgende Schlange sein:
GGGAGCGUAGCUCAGUGCGGAGAGCGCCUGCUUUGCACGCAGGAGGUCUGCGGUUCGAUC CCGCGCGCUCCCACCA

2. Eine Schlängelung mit möglichst vielen Harmoniestellen wäre der Ribo-Natter am liebsten. Beschreibe, wie eine Schlange vorgehen könnte, um die maximale (Farb-)Harmonie zu finden.

■ AUFGABE 2: WO BIN ICH?

Labormaus Ludwig erwacht aus der Narkose. Um sich herum fühlt er Sperrholzwände, und er weiß: Wieder hat die nette Verhaltensforscherin ihn irgendwo in das Labyrinth gesetzt, das er nun schon so gut kennt. Immer an derselben Stelle steht ein Napf mit etwas Futter. Die Verhaltensforscherin wird ihm außerdem noch mehr Futter geben, wenn er am Napf ist, und zwar umso mehr, je eher er dort ist – so hat Ludwig sie inzwischen dressiert.

Ludwig ist hungrig wie immer, und er weiß genau, was er tun muss: Am schnellsten kommt er zum Futter, wenn er zunächst herausfindet, wo exakt er im Labyrinth ist, und von dort den kürzesten Weg zum Futternapf rennt. Da es immer dasselbe Labyrinth ist, hat er die Wegekarte klar im Kopf. (Ludwig entstammt einer alten Dynastie von Labormäusen, die über viele Generationen gelernt haben, sich Labyrinthe einzuprägen.) Er kann ferner das Rauschen der Klimaanlage an der Nordwand des Laborraums hören, so dass er jederzeit weiß, wo Norden liegt. Da es dunkel ist im Labyrinth, kann er leider nur Wände unmittelbar um ihn herum wahrnehmen und nicht in Gänge oder Abzweige hineinschauen.

Seine Position im Labyrinth bestimmt Ludwig so: Er läuft die Labyrinthgänge entlang zu Kreuzungen, Einmündungen, Ecken und Sackgassen so lange, bis er an den Ganglängen und an der Form und Lage all der passierten Kreuzungen nach seiner Karte im Kopf eindeutig erkennen kann, wo er sich befindet. Manchmal kommt er zufällig schon während der Positionsbestimmung am Futternapf an: Das spart Lauferei, denn dann kann er direkt fressen.

Das Labyrinth ist abgeschlossen, Ludwig kann es also nicht verlassen. Alle Orte darin sind von allen Orten aus zu erreichen. Die Gänge des Labyrinths laufen in die Richtungen Nord, Ost, Süd oder West. Manche Gänge enden in Sackgassen. Die Futterstelle liegt am Ende einer Sackgasse.

» Aufgabe

Die Positionen in einem Labyrinth liegen auf einem Raster mit Spalten und Zeilen, die zur einheitlichen Benennung mit Zahlen markiert sind. Die folgende Abbildung zeigt ein Labyrinth mit je 15 Spalten und Zeilen; der Futternapf steht an Position (3,10).

Schreibe ein Programm, das Ludwigs Methode zur Positionsbestimmung nachvollzieht. Gib das Verhalten der Maus Ludwig an drei Beispielen mit zufällig gewählten Startpunkten wieder. Das abgebildete Labyrinth muss mindestens in einem der Beispiele verwendet werden; dabei soll sich der Startpunkt an Position (14,3) befinden.

■ AUFGABE 3: GUTE STUBE

Die Cyber-Künstlerin EKAI will all die öden Chatrooms möblieren. Sie hat schon hübsche Bilder für die Objekttypen „Sessel“, „Couch“, „Zimmerpflanze“, „Kunstobjekt“, „Sitzkissen“, „Fahrrad“, „Lagerfeuer“, „Teddybär“, „Stehlampe“ und „Haustier“ entworfen. EKAI will, dass Chatter die vorkommenden Gegenstände und ihre Lage zueinander gemeinsam beschreiben und dann alle das Foto ihres Chatrooms auf dem Bildschirm sehen können. Die möglichen Lagen zueinander seien „links von“, „rechts von“, „neben“, „vor“, „hinter“, „über“ und „unter“.

Aufgabe

1. Schreibe für EKAI ein Programm, das
   a) aus einer Folge von Sätzen die gemeinten Gegenstände und Lagebeschreibungen herausfiltert,
   b) alle Mehrdeutigkeiten, Unvollständigkeiten und Widersprüche in der Gesamtbeschreibung selbstständig irgendwie entscheidet und
   c) ein 400x300 Pixel großes Bitmap (Foto) mit einer grafischen Umsetzung der Gesamtbeschreibung erzeugt (ob in Schwarzweiß oder Farbe, ist uns egal). Benutze dazu digitale grafische Darstellungen (Grafiken, Fotos) für die Objekttypen von EKAI.

2. Erläutere kurz, welche Mehrdeutigkeiten, Unvollständigkeiten und Widersprüche dein Programm in Gesamtbeschreibungen erkennen kann und nach welchen Prinzipien es sie entscheidet. Diese Prinzipien müssen nicht begründet werden – künstlerische Freiheit ist hier sowieso unvermeidlich.

3. Schicke uns drei Satzfolgen (als unformatierte Textdateien) und die dazu erzeugten Bitmaps (als GIF-, JPEG- oder BMP-Dateien). Eine der drei Satzfolgen sei:
   - Ich hätte gerne das Fahrrad#1 links von Stehlampe#1.
   - Der Sessel#1 steht links von Sessel#2.
   - Couch#1 befindet sich vor Zimmerpflanze#2.
   - Sessel#1 ist neben dem Lagerfeuer#1.
   - Ein Sessel#2 steht hinter Lagerfeuer#1.
   - Sitzkissen#1 unter Haustier#1.
   - Kunstobjekt#1 schwebt über Lagerfeuer#1.
   - Die Zimmerpflanze#1 hinter Teddybär#1.
   - Fahrrad#1 ist irgendwo rechts von Sessel#2.
   - Stehlampe#1 sieht man links von Sessel#1.
   - Doch die Zimmerpflanze#2 steht hinter dem Fahrrad#1.

■ AUFGABE 4: VERSTEHST DU BAHNHOF?

Jeden Tag treffen ein Güterzug aus Produdorf und einer aus Herställen in Mittelstadt ein, wo sie zu zwei neuen Zügen zusammengestellt werden, von denen einer nach Konsuweiler und einer nach Verberau fährt. Jeder Zug besteht aus einer Lok gefolgt von Waggons. Du möchtest dem Bahnhofsvorsteher von Mittelstadt helfen, das Auseinander- und Zusammenkoppeln der Waggons zu planen.

In Mittelstadt werden die Züge für Konsuweiler und Verberau aus den eintreffenden Zügen entsprechend dem gemeldeten Tagesbedarf zusammengestellt. Das Auseinander- und Zusammenkoppeln der Waggons ist aufwändig und zeitraubend. Daher sollte die Anzahl der Koppelvorgänge minimiert werden, wobei ein Koppelvorgang zwei Waggons bzw. eine Lok und einen Waggon auseinander- oder zusammenkoppelt.

Aufgabe

Schreibe ein Programm, das den Tagesbedarf von Konsuweiler und den von Verberau sowie die in Mittelstadt eintreffenden Züge als Folgen von Waggontypen einliest und eine Folge von Koppelanweisungen ausgibt, mit der die beiden in Mittelstadt eintreffenden Züge mit möglichst wenig Koppelvorgängen zu den beiden abfahrenden Zügen zusammengestellt werden können.

Zug aus Produdorf:

Zug aus Herställen:

Die Waggons sind vom Typ A, B, C usw., je nachdem, ob sie Autoklaven, Bumerange, Celli usw. transportieren. Für jeden Typ wird der Tagesbedarf (eine ganze Anzahl Waggons) von Konsuweiler und ebenso der von Verberau morgens den anderen Orten telefonisch mitgeteilt, und es wird dafür gesorgt, dass die Züge aus Produdorf und Herställen zusammen von jedem Typ genau die insgesamt geforderte Anzahl an Waggons enthalten. Ferner kann man davon ausgehen, dass in jedem dieser Züge alle Waggons eines Typs unmittelbar aufeinander folgen, wogegen das für die aus Mittelstadt abfahrenden Züge nicht gefordert wird.

In Mittelstadt stehen hinreichend viele Rangiergleise und Rangierloks zur Verfügung, deren Benutzung „gratis“ ist (Koppelvorgänge zwischen einer Rangierlok und einem Waggon werden nicht gezählt). Die aus Produdorf und Herställen ankommenden Loks müssen aber nach Konsuweiler und Verberau weitergeführt werden.

Teste dein Programm anhand von drei Eingaben. Eine davon soll aus dem unten angegebenen Tagesbedarf und den oben abgebildeten Zügen bestehen.

Tagesbedarf an:	A	B	C	D	E
Konsuweiler:	5	0	7	8	3
Verberau:	7	6	8	0	4

■ AUFGABE 5: DOSEN PACKEN

Ein Industrieroboter an einem Fließband muss so programmiert werden, dass er möglichst viele ankommende zylinderförmige Dosen in einer Schicht in eine 1 m x 1 m große Kiste packt. Die Durchmesser der Dosen sind unterschiedlich und liegen zwischen 10 cm und 20 cm. Der Roboter muss die Dosen der Reihe nach einpacken.

Aufgabe

- Schreibe eine Funktion „verschiebe“, die eine teilweise gefüllte Kiste, die Position einer Dose in der Kiste (x- und y-Koordinaten) und eine Richtung (einen Winkel) als Eingabe nimmt und die Dose so lange in die gegebene Richtung verschiebt, bis sie an eine schon platzierte Dose oder an den Rand der Kiste stößt.

  Die abschließende Position der Dose soll ausgegeben werden. Demonstriere die Funktion an drei Beispielen.

- Überlege dir und beschreibe in einem kurzen Text mindestens zwei Methoden, die die Funktion „verschiebe“ verwenden, um eine möglichst gute Packung zu erzielen.

Bitte in Druckschrift ausfüllen und Zutreffendes ankreuzen. ☒

Geburtsdatum . . . . . Name, Vorname

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 ... ☐

Programmiersprache? Betriebssystem? Computer(typ)? eigener

Hast du an anderen Wettbewerben teilgenommen? ☐ Informatik ☐ Mathematik ☐ sonstige

Wie hast du vom Wettbewerb erfahren? ☐ 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