



## 26. Bundeswettbewerb Informatik 2007/2008

# Die Aufgaben der zweiten Runde

### Allgemeine Hinweise

An dieser Runde dürfen nur Einzelpersonen teilnehmen, die in der ersten Runde in drei Aufgaben mindestens 12 Punkte erreicht oder einer Gruppe angehört haben, der dieses gelungen ist. Gruppenarbeit ist in der zweiten Runde nicht zulässig.

**Einsendeschluss ist der 7. April 2008;** es gilt das Datum des Poststempels. Bitte sende deine Lösungen wieder an den **Bundeswettbewerb Informatik, Ahrstraße 45, 53175 Bonn.**

Es gibt drei Aufgaben. **Wichtig:** Eine Einsendung darf nur Bearbeitungen zu höchstens zwei Aufgaben enthalten, deren Bewertung dann das Gesamtergebnis ausmacht. Sollte eine Einsendung Bearbeitungen zu allen drei Aufgaben enthalten, werden wir zwei davon zufällig auswählen und nur diese bewerten.

Die Bearbeitung einer Aufgabe sollte zunächst eine einfache, nachvollziehbare und vollständige Lösung aller Teilaufgaben enthalten. **Pluspunkte** für eine höhere Bewertung kannst du erreichen, wenn du die Aufgabe dort, wo es möglich und sinnvoll ist, eigenständig weiterentwickelst. Dabei bevorzugen wir inhaltliche Erweiterungen und Verbesserungen, etwa von Datenstrukturen und Algorithmen, und legen keinen Wert auf aufwändige Tricks, z.B. zur reinen Verschönerung der Bedienungsoberfläche. Begründe für jede Erweiterung, weshalb sie sinnvoll ist und ihre Realisierung eine eigene Schwierigkeit darstellt.

Denke bitte daran, dass zur Bewertung möglicherweise nur die Papierunterlagen herangezogen werden können. Diese sollten also einen lückenlosen und nachvollziehbaren Nachweis des Leistungsumfangs und der Funktionstüchtigkeit der Programme geben. Der Umfang der Einsendung soll sich in Grenzen halten; eine gute Dokumentation vermittelt kurz und präzise alles Nötige, insbesondere die wesentlichen Lösungsideen. Nötig ist alles, was Interessierte mit guten Informatikkenntnissen, die die Aufgabenstellung kennen, wissen müssen, um die Lösungsidee zu verstehen und die Realisierung dieser Idee nachzuvollziehen. Generell sind gute und originelle Ideen zwar entscheidend, aber die Dokumentation hat schon oft den Ausschlag für oder gegen das Weiterkommen gegeben.

Grundsätzlich gelten die Gliederungs- und Dokumentationsrichtlinien der 1. Runde weiter. Zu jeder Teilaufgabe gehört also die Lösungsidee und die Dokumentation der Lösung sowie

des dazu gehörigen Programms (eine Beschreibung, wie die Idee in konkrete Programmelemente umgesetzt wurde, Hinweise auf Nutzungsgrenzen, Besonderheiten usw.). Dabei sind (halb-)formale Notationen besser als Programmausschnitte. Für die geforderten Programme erwarten wir Programmablaufprotokolle, also kommentierte Probeläufe des Programms, aus denen ersichtlich wird, wie das Programm sich in unterschiedlichen Situationen verhält. Sende uns außerdem bitte (abgedruckt!) aussagekräftige Ergebnisse von Programmläufen mit unterschiedlichen Daten. Komplettiert wird das Papiermaterial durch den Programmtext, wobei unwichtige und automatisch generierte Teile nicht ausgedruckt werden sollen.

Schicke uns alles in lesbarer Form auf Papier, Schriftgröße mindestens 10 Punkt, bei Quelltext mindestens 8 Punkt. Bitte schicke uns deine Unterlagen auf losen, gelochten Blättern im Format DIN A4 (Hüllen mit Lochrand nur bei ausreichender Stabilität verwenden; keine Heftstreifen oder Mappen) und gib auf jedem Blatt Verwaltungsnummer, Vorname, Name und Seitennummer an. Die Verwaltungsnummer steht auf der Teilnahmebescheinigung der ersten Runde. Bitte gliedere deine Einsendung in (a) Allgemeines, (b) Unterlagen zur ersten bearbeiteten Aufgabe und (c) Unterlagen zur zweiten bearbeiteten Aufgabe.

Außerdem sende uns bitte die Programmtexte und lauffähigen Programme auf einer CD oder DVD. Bei der Bewertung können Programme unter Windows (XP), Linux und MacOS X (10.4) ausgeführt werden.

Fragen zu den Aufgaben dürfen per E-Mail an [bwinf@bwinf.de](mailto:bwinf@bwinf.de) oder telefonisch unter 0228-378646 (zu üblichen Arbeitszeiten) gestellt werden. Die Antwort auf E-Mail-Anfragen kann sich leicht verzögern. Informationen zur 2. Runde finden sich auf den Webseiten des BWINF ([www.bwinf.de](http://www.bwinf.de)). In der newsgroup [fidonet@bwinf.de](mailto:fidonet@bwinf.de) wird sicher wieder über die Aufgaben diskutiert werden – ohne Lösungsideen auszutauschen.

Allen Teilnehmern der zweiten Runde wird Anfang Juni die Bewertung mitgeteilt. Die Besten werden zur Endrunde eingeladen, die im Herbst 2008 in Saarbrücken stattfinden wird, ausgerichtet vom Max-Planck-Institut für Informatik, dem Max-Planck-Institut für Softwaresysteme, dem Deutschen Forschungszentrum für Künstliche Intelligenz und der Universität des Saarlandes. Dort werden die Bundessieger und Preisträger ermittelt und am letzten Tag ausgezeichnet. Bundessiegern winkt die Aufnahme in die Studienstiftung des deutschen Volkes. Außerdem werden Geld- und Sachpreise vergeben. Der Rechtsweg ist ausgeschlossen.

Die folgenden Informationen können vielleicht zusätzlich dazu motivieren, sich an der zweiten Runde aktiv zu beteiligen: Es ist prinzipiell möglich, eine Einreichung zur zweiten Runde als so genannte besondere Lernleistung in die Abiturwertung einfließen zu lassen; Genaueres erfährst du an deiner Schule. Jüngere Zweitrundenteilnehmer haben die Chance, zu einer Schülerakademie eingeladen zu werden. Und sehr gute Leistungen, die nur knapp nicht zur Qualifikation für die Endrunde ausgereicht haben, werden mit Buchpreisen des Verlags O'Reilly prämiert.

Viel Spaß und viel Erfolg!

## MCI-Sonderpreis

Beim Bundeswettbewerb Informatik spielt die Mensch-Computer-Schnittstelle oder Bedienschnittstelle eines eingesandten Programms bei der Bewertung prinzipiell keine Rolle. Für die Benutzbarkeit von Informatiksystemen ist diese Komponente aber von ganz entscheidender Bedeutung, und so wird bei einigen Einsendungen zum BWINF erhebliche Mühe auf den Interaktionsaspekt verwendet.

Diese Mühe soll belohnt werden: Der Fachbereich Mensch-Computer-Interaktion (MCI) der Gesellschaft für Informatik (GI) schreibt erneut einen Sonderpreis für besonders gelungene Bedienschnittstellen aus. Verliehen wird dieser Preis auf der Tagung „Mensch & Computer 2008“, die vom 7. bis 10. September 2008 in Lübeck stattfinden wird.

Du hast die Möglichkeit, dich mit einem der von dir eingesandten Programme um den MCI-Sonderpreis zu bewerben. Beschreibe dazu in einem separaten Dokument die Bedienschnittstelle dieses Programms. Gib aber nicht nur eine Bedienungsanleitung, sondern erlautere vor allem Entwurfskonzept und -entscheidungen.

Das Programm und seine Bedienschnittstelle werden nach folgenden Kriterien bewertet (auf die dein Entwurfskonzept eingehen sollte):

**Dialogkriterien:** Der Dialog eines Benutzers mit einem System ist

*aufgabenangemessen*, wenn das System den Benutzer bei der Ausführung der Aufgaben unterstützt, d.h. bei seinen Arbeitsschritten sinnvoll führt, ohne Handlungen unnötig einzuschränken;

*selbsterklärend*, wenn jeder Dialogschritt durch Reaktionen des Systems sofort verständlich wird oder dem Benutzer auf Anforderung erklärt wird;

*kontrollierbar*, wenn das System in jeder Situation auf die BenutzerInnen reagiert und diese agieren, d.h. Richtung und Geschwindigkeit der Interaktion bestimmen können, bis ihr Ziel erreicht ist;

*fehlertolerant*, wenn trotz offensichtlicher Fehler in der Eingabe das beabsichtigte Ergebnis ohne bzw. mit nur geringem Eingreifen der Benutzer erzielt werden kann.

**Präsentationskriterien:** Die Präsentation von Information (also z.B. auch der Bezeichnung von Dialogelementen) ist

*fachlich gut gestaltet*, wenn zusammengehörige Informationen räumlich gruppiert sind, ein „aufgeräumter“ Eindruck entsteht;

*übersichtlich*, wenn die Menge der Informationen knapp und strukturiert dargestellt wird;

*lesbar, klar und präzise*, wenn die (visuelle) Darstellung der Information leicht zu lesen bzw. erkennen ist und der Informationsgehalt schnell und genau und ohne überflüssige Informationen vermittelt wird.

**BWINF-Kriterien:** Ein Programm und seine Bedienschnittstelle sind

*originell*, wenn die Bedienschnittstelle ungewöhnlich und mit eigenen Mitteln (aber dennoch ergonomisch) gestaltet ist;

*inspizierbar*, wenn die Bedienschnittstelle vollen Zugang zur Funktionalität erlaubt, also z.B. Kontrollflüsse dargestellt werden oder Parameter beeinflusst werden können.

## Aufgabe 1: Berufsinformationstag

Ludwig Lämpel ist Lehrer für Gemeinschaftskunde und Biologie sowie Koordinator für Berufsorientierung an der Wilhelm-Busch-Schule. Traditionell organisiert er jährlich einen Berufsinformationstag. Dafür konnte er diesmal 16 Elternteile gewinnen, die über ihre Berufe in der Schule berichten und Fragen der Schülerinnen und Schüler beantworten wollen. Der Berufsinformationstag soll ein Samstag sein; daher ist die Teilnahme freiwillig, jedoch nach Anmeldung verbindlich. Nach Erfahrungen der letzten Jahre ist die Beteiligung sehr unterschiedlich, wird aber zwischen 80 und 240 Teilnehmenden liegen.

Der Vormittag soll wie folgt organisiert werden:

9:00- 9:25	Begrüßung durch die Schulleitung in der Aula
9:30-10:15	Berufsinformation in parallelen Veranstaltungen
10:20-11:05	Berufsinformation in parallelen Veranstaltungen
11:05-11:20	Pause
11:20-12:05	Berufsinformation in parallelen Veranstaltungen
12:10-12:55	Berufsinformation in parallelen Veranstaltungen

Die Schülerinnen und Schüler können an bis zu vier der 16 Berufsinformationsveranstaltungen teilnehmen. Bei der Anmeldung geben sie bis zu sechs Wünsche (davon zwei Ersatzwünsche) in der Reihenfolge ihrer Priorität an. Die Zuordnung auf die einzelnen Veranstaltungen erfolgt durch den Organisator. Alle Eltern haben sich bereit erklärt, je nach Nachfrage eine Veranstaltung auch mehrmals nacheinander durchzuführen.

Für die Durchführung einer Veranstaltung ist eine Mindest- und eine maximale Teilnehmerzahl einzuhalten. Die maximale Teilnehmerzahl kann je nach Veranstaltung variieren, weil sich die Eltern unterschiedliche methodische Gestaltungen überlegt haben. Weil Herr Lämpel in den letzten Jahren zu viel Zeit für die Organisation benötigte, hat er sich kürzlich seinen ersten Computer gekauft. Er benötigt jetzt „nur noch“ ein leicht zu benutzendes Programm, das ihn bei der Planung des Berufsinformationstages unterstützt.

### Aufgabe

1. Entwickle und beschreibe Kriterien, mit denen beurteilt werden kann, wie oft eine bestimmte Veranstaltung angeboten und wie die Teilnehmenden den Veranstaltungen zugeordnet werden sollten. Erstelle mehrere Vorschläge und formuliere dazu sinnvolle Annahmen über die unterschiedlichen Interessen der Beteiligten.
2. Identifiziere die Bedingungen, die erfüllt sein müssen, damit eine bestimmte Zuordnung von Schülerinnen und Schülern auf Veranstaltungen den oben beschriebenen Vorgaben genügt, also zulässig ist.
3. Implementiere ein Programm, das Herrn Lämpel bei der Festlegung der Veranstaltungen und der Zuordnung von Teilnehmenden auf die Veranstaltungen unterstützt und die Ergebnisse aus (1) und (2) berücksichtigt. Verwende dazu als eine Komponente ein Werkzeug zur ganzzahligen linearen Programmierung wie etwa das GNU Linear Programming Kit (GLPK). Hinweise zu dessen Installation und Verwendung findest du auf der Webseite des BWINF.

## Aufgabe 2: Blocksberg

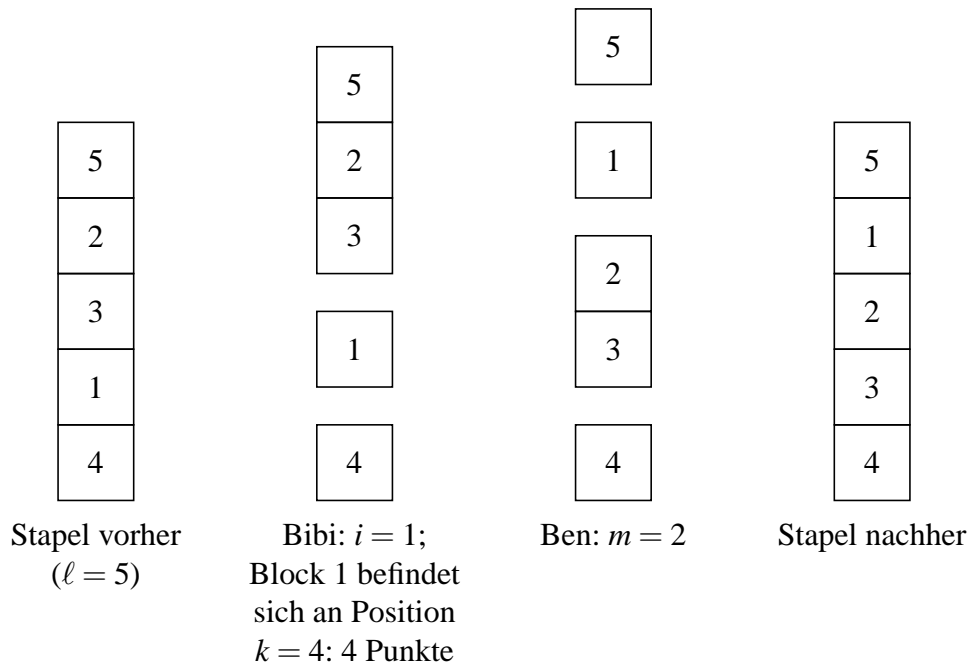


Abbildung 1: Ein Spielzug in *Blocksberg* (Bibi liest ihre Liste vor)

Der Entwurf des Zweipersonenspiels *Blocksberg* nähert sich dem Abschluss. Für eine kleine natürliche Zahl  $\ell$  befindet sich auf einem Tisch zwischen den beiden Spielern – nennen wir sie Bibi und Ben – ein Stapel von  $\ell$  Blöcken, die von allen Seiten gut sichtbar mit den Nummern  $1, \dots, \ell$  beschriftet sind. Bibi schreibt zuerst auf einem Blatt Papier, ohne dass Ben etwas davon sieht, eine Folge von Zahlen aus  $\{1, \dots, \ell\}$ , wobei sie jede Zahl beliebig oft benutzen darf. Dann liest sie die Zahlen ihrer Liste der Reihe nach laut vor. Nachdem Bibi eine Zahl  $i$  gelesen hat, wird der Block mit Nummer  $i$  gesucht. Wenn dieser momentan der  $k$ te Block von oben im Stapel ist, gewinnt Bibi  $k$  Punkte. Danach muss Ben mit einer Hand die  $k - 1$  Blöcke oberhalb des Blocks mit Nummer  $i$  hochheben, den Block mit Nummer  $i$  mit der anderen Hand herausnehmen und dann die  $k - 1$  Blöcke zwischen den Fingern auf den Stapel zurückgleiten lassen, wobei er den Block mit Nummer  $i$  an irgendeiner Stelle – er entscheidet, wo – dazupackt. Der Block mit Nummer  $i$  ist also danach der  $m$ te Block von oben, für eine von Ben frei wählbare ganze Zahl  $m$  mit  $1 \leq m \leq k$ , während die anderen  $k - 1$  hochgehobenen Blöcke ihre Reihenfolge beibehalten.

Abbildung 1 verdeutlicht die Abfolge der Handlungen eines Spielzugs in *Blocksberg*.

Nachdem Ben Bibis ganze Liste abgearbeitet hat und sie dabei tüchtig Punkte sammeln konnte, ist sie selbst dran. Ben schreibt aber keine neue Liste, sondern Bibi muss ihre eigene Liste in derselben Weise abarbeiten, während Ben wie vorher Bibi Punkte sammelt.

**Aufgabe**

1. Die obige Beschreibung ist unvollständig und lässt gewisse Aspekte des Spiels offen. Finde solche unzureichend beschriebene Stellen, wähle jeweils eine Festlegung der Spielregeln und beschreibe und begründe deine Wahl.
2. Schreibe Programmkomponenten, die sich wie Bibi und Ben verhalten, und die zusammenarbeiten können, um ein Spiel durchzuführen. Bibi soll zuerst versuchen, eine möglichst „fiese“ Liste zu erstellen, um viele Punkte zu sammeln, während Ben immer wieder versuchen soll, die Zahl  $m$  möglichst schlau zu wählen, damit Bibi doch nicht so viele Punkte bekommt. Danach soll Bibi die Liste allein abarbeiten und dabei Ben möglichst wenig Punkte „schenken“. Dein Programm soll als Minimum die benutzte Liste und die Anzahl der erworbenen Punkte von Bibi und Ben ausgeben; interessantere Ausgaben sind erwünscht. Beschreibe die Spielstrategien, die du für Bibi und Ben gewählt hast, und kommentiere, ob sie gut zu funktionieren scheinen.
3. Es ist klar, dass das Spiel nicht ganz ausgeglichen ist: Ben tappt beim Spielen immer im Dunkeln, da er nie weiß, welcher Block als Nächster angefordert wird, während Bibi von vornherein die ganze Liste kennt. Aus diesem Grund wird Bibi fast immer mehr Punkte als Ben sammeln können. Deswegen wird vereinbart, dass Bens Gesamtpunktezahl mit einer Konstante  $c > 1$  multipliziert werden soll, bevor der Spieler mit der höchsten Gesamtpunktezahl zum Gewinner erklärt wird. Die Wahl von  $c$  steht aber aus; das ist etwas, was an dem Entwurf des Spiels noch fehlt. Beschreibe und motiviere Experimente, die Aufschluss darüber geben können, welche Werte von  $c$  das Spiel besonders fair werden lassen. Führe solche Experimente durch, dokumentiere sie und mache zum Schluss einen begründeten Vorschlag für die Wahl von  $c$ .

## Geheimnisvolle Töne

Mit einem Funkteleskop konnte neulich ein Tonsignal empfangen und aufgezeichnet werden, dessen Struktur eine natürliche Herkunft so gut wie ausschließt. Es stellte sich heraus, dass das Signal höchst wahrscheinlich irdischen Ursprungs ist und kodierte Information enthält.

Zu Beginn ist die Qualität der mitgeschnittenen Aufnahme sehr gut, während sie zum Ende hin immer mehr verrauscht ist. Das Signal steht der Öffentlichkeit in Form einer Audiodatei im WAVE-Format (Dateiendung: `.wav`) auf der Webseite des BWINF zur Verfügung.

### Aufgabe

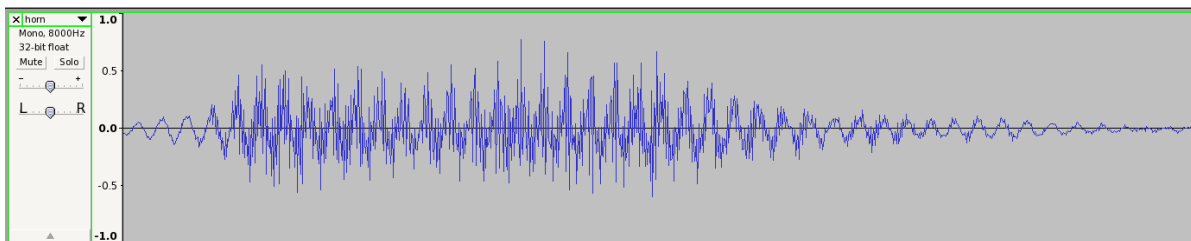
Nutze Deine detektivischen Fähigkeiten, um das Signal zu analysieren und Regelmäßigkeiten zu entdecken, die auf die verwendete Kodierung schließen lassen. Verlasse dich dabei nicht nur auf dein Ohr, sondern setze auch deinen Computer ein.

Erstelle ein Programm, das die verborgene Botschaft und auch andere Nachrichten dieser Form aufdecken kann. Es könnte von Vorteil sein, das Programm schrittweise zu erweitern und aus den Ergebnissen jedes Schrittes zu lernen.

### WAVE-Dateien

Ein gehörter Ton ist, vereinfacht gesagt, eine vibrierende Luftsäule, deren Bewegung sich im Ohr auf das Trommelfell überträgt. Ein Ton kann graphisch dargestellt werden, indem wir auf der  $y$ -Achse den räumlichen Ausschlag des Trommelfells aus seiner Ruheposition und auf der  $x$ -Achse die Zeit eintragen. Ein Ton kann elektronisch übertragen werden, indem der Ausschlag der Membrane (die Amplitude) eines Mikrophons in eine Spannung übersetzt wird. Dies kann beispielsweise direkt durch einen Magneten, der sich in einer Spule bewegt, erreicht werden, oder indirekt, beispielsweise durch einen Kondensator, dessen Kapazität sich mit der Amplitude ändert. Ein Ton kann digital übertragen werden, indem die Amplitude durch eine Binärzahl angenähert wird. Natürlich kann eine solche Binärzahl nicht für *jeden* Zeitpunkt übertragen werden: Wir wählen normalerweise diskrete und äquidistante Zeitpunkte.

Das folgende Bild zeigt den Ton einer Autohupe, welcher mit 8000 Hz abgetastet wurde und dessen Amplitude in nur 8 Bits dargestellt wird:



Das Shannonsche Abtasttheorem besagt, dass für Töne mit einer Frequenz von bis zu 20000 Hz eine Abtastrate von mindestens 40000 Hz gewählt werden muss, damit durch den Diskretisierungsfehler kein Qualitätsverlust eintritt. Für CDs wurde eine Abtastrate von 44100 gewählt, während die Amplitude durch ein 16-Bit-Wort dargestellt wird.

Die dieser Aufgabe zu Grunde liegenden Töne sind in CD-Qualität gespeichert. Die sie beinhaltende Datei ist im beliebten WAVE-Format, welches in unserem Fall aus einem Header aus 44 Bytes besteht, gefolgt von 6948253 Worten aus 16 Bits – die Amplituden in Zweierkomplementdarstellung. Jedes dieser Worte besteht aus zwei Bytes: dem höherwertigen Byte mit den oberen 8 Bits und dem niederwertigen Byte mit den unteren 8 Bits. Vorsicht: In der Datei stehen die Worte zwar in der richtigen Reihenfolge, aber zu jedem einzelnen Wort kommt der niederwertige Teil zuerst, direkt gefolgt vom höherwertigen Teil („Little Endian“).

Für Interessierte gibt es weitere Informationen zum WAVE-Format in der Wikipedia<sup>1</sup> und an vielen anderen Stellen im WWW. In der Wikipedia gibt es auch interessante Informationen zur Signalverarbeitung<sup>2</sup>, die allerdings für die Bearbeitung dieser Aufgabe *nicht* benötigt werden.

---

<sup>1</sup>[de.wikipedia.org/wiki/RIFF\\_WAVE](http://de.wikipedia.org/wiki/RIFF_WAVE)

<sup>2</sup>[en.wikipedia.org/wiki/Signal\\_processing](http://en.wikipedia.org/wiki/Signal_processing)