



## 23. Bundeswettbewerb Informatik 2004/2005

# Die Aufgaben der zweiten Runde

### Allgemeine Hinweise

An dieser Runde können nur Einzelpersonen teilnehmen, die in der ersten Runde in drei Aufgaben mindestens 12 Punkte erreicht haben oder einer Gruppe angehört haben, der dieses gelungen ist. Gruppenarbeit ist in der zweiten Runde nicht zulässig. **Einsendeschluss ist der 18. April 2005**, wobei das Datum des Poststempels gilt. Bitte adressieren Sie Ihre Einsendung wieder an den **Bundeswettbewerb Informatik, Ahrstraße 45, 53175 Bonn**.

Es gibt drei Aufgaben. **Wichtig:** Sie dürfen nur Bearbeitungen zu höchstens zwei Aufgaben einsenden, deren Bewertung dann Ihr Gesamtergebnis ausmacht. Sollten Sie uns doch Bearbeitungen zu allen drei Aufgaben schicken, 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 können Sie erreichen, wenn Sie die Aufgabe dort, wo es möglich und sinnvoll ist, eigenständig weiterentwickeln. 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 Bedienungs Oberfläche. Begründen Sie für jede Erweiterung, weshalb sie sinnvoll ist und ihre Realisierung eine eigene Schwierigkeit darstellt.

Denken Sie bitte daran, dass zur Bewertung möglicherweise nur Ihre Papier-Unterlagen herangezogen werden können. Diese sollten also einen lückenlosen und nachvollziehbaren Nachweis des Leistungsumfangs und der Funktionstüchtigkeit Ihrer Programme geben. Dem Umfang Ihrer Einsendung setzen Sie bitte Grenzen; eine gute Dokumentation vermittelt kurz und präzise alles Nötige, insbesondere die wesentlichen Lösungsideen. Nötig ist alles, was Interessierte mit guten Informatikkenntnissen, die nur die Aufgabenstellung kennen, wissen müssen, um Ihre Lösungsidee zu verstehen und Ihre 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 dazugehörigen Programms (eine Beschreibung, wie die Idee z.B. 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. Senden Sie uns außerdem bitte aussagekräftige Ergebnisse von Programmläufen mit unterschiedlichen Daten. Komplettiert wird das Papier-Material durch den Programmtext, wobei unwichtige und automatisch generierte Teile nicht ausgedruckt werden sollen.

Schicken Sie uns alles in lesbarer Form auf Papier, Schriftgröße mindestens 10 Punkt, bei Quelltext mindestens 8 Punkt. Bitte schicken Sie uns Ihre Unterlagen auf losen, gelochten Blättern im Format DIN A 4 (Hüllen mit Lochrand nur bei ausreichender Stabilität verwenden; keine Heftstreifen oder Mappen) und geben Sie auf jedem Blatt Verwaltungsnummer, Vorname, Name und Seitennummer an. Sie finden Ihre Verwaltungsnummer auf der Teilnahmebescheinigung der ersten Runde. Bitte gliedern Sie Ihre Einsendung in (a) Allgemeines, (b) Unterlagen zur ersten bearbeiteten Aufgabe und (c) Unterlagen zur zweiten bearbeiteten Aufgabe.

Außerdem senden Sie uns bitte die Programmtexte und lauffähigen Programme auf einer CD oder auch 3,5“-Diskette(n) (lesbar unter DOS/Windows98; die Programme dürfen natürlich auch für andere Betriebssysteme entwickelt worden sein).

Für Fragen zu den Aufgaben gibt es wieder eine Hotline, und zwar unter 0228 - 37 86 46 jeweils von 17-19h am 11.1., 2.2., 1.3. und 5.4.2005 oder unter [bwinf@bwinf.de](mailto:bwinf@bwinf.de). Die Antwort auf E-mail-Anfragen kann sich leicht verzögern. Informationen zur 2. Runde finden sich auf unseren Webseiten ([www.bwinf.de](http://www.bwinf.de)). In der newsgroup [fido.ger.bwinf](mailto:fido.ger.bwinf) 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 vom 21. bis 24. September 2005 vom Hasso-Plattner-Institut in Potsdam mit Unterstützung der SAP AG ausgerichtet werden wird. 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 Sie vielleicht zusätzlich motivieren, sich an der zweiten Runde aktiv zu beteiligen: In vielen Bundesländern gibt es Anerkennungen für gute Teilnehmer an der zweiten BWINF-Runde. Außerdem ist prinzipiell möglich, eine Einreichung zur zweiten Runde als so genannte besondere Lernleistung in die Abiturwertung einfließen zu lassen. Genaueres erfahren Sie an Ihrer 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 Interaktionskomponente 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: Zum fünften Mal schreibt der Bundeswettbewerb Informatik gemeinsam mit dem Fachbereich Mensch-Computer-Interaktion (MCI) der Gesellschaft für Informatik (GI) einen Sonderpreis aus für besonders gelungene Interaktionskomponenten. Die beste Leistung wird mit einem Sachpreis belohnt. Verliehen wird dieser Preis auf der Tagung „Mensch & Computer 2005“, die vom 4. bis 7. September 2005 im österreichischen Linz stattfindet.

Sie können sich mit einem der von Ihnen eingesandten Programme um den MCI-Sonderpreis bewerben. Beschreiben Sie dazu in einem separaten Dokument die Interaktionskomponente dieses Programms. Geben Sie aber nicht nur eine Bedienungsanleitung, sondern erläutern Sie vor allem Entwurfskonzept und -entscheidungen.

Das Programm und seine Interaktionskomponente werden an Hand einer Reihe von Kriterien bewertet, die unten näher erklärt sind. Sie richten sich mehrheitlich nach einigen Dialogprinzipien und wichtigen Eigenschaften der Informationspräsentation, die in Normen für Mensch-Computer-Schnittstellen gefordert werden. Dazu kommen zwei Aspekte, die für den BWINF wichtig sind. Im einzelnen handelt es sich um folgende Kriterien:

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

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

*kontrollierbar*, wenn Benutzer 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 ist

*klar und präzise*, wenn der Informationsgehalt schnell und genau und ohne überflüssige Informationen vermittelt wird;

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

*lesbar*, wenn die (visuelle) Darstellung der Information leicht zu lesen bzw. erkennen ist.

**BWINF-Kriterien:** Ein Programm und seine Interaktionskomponente ist

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

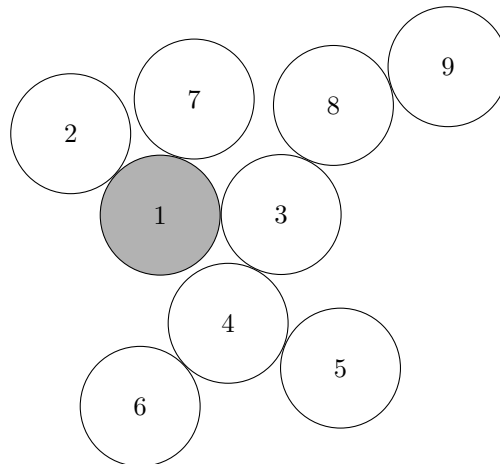
*inspizierbar*, wenn die Interaktionskomponente Zugang zur Funktionalität erlaubt und die Bewertung des Programms erleichtert.

## Aufgabe 1: Klebende Kugeln

In der Natur trifft man oft auf bizarre Formen und fragt sich, wie sie entstehen können. Oft spielen kleine Unregelmäßigkeiten, die zufällig entstanden sind, dabei eine Rolle. Sie werden anschließend durch selbstverstärkende Effekte immer größer. In einer Simulation soll ein einfacher Wachstumsprozess nachgebildet werden, der diesen Effekt besonders schön demonstriert:

Auf einer unbeschränkt großen Ebene klebt anfangs an einer Stelle eine Kugel. Nun kommen immer wieder Kugeln derselben Größe aus zufälligen Richtungen angerollt. Trifft zum ersten Mal eine solche auf die Urkugel, dann bleibt sie fest an ihr kleben und es entsteht ein Gebilde aus zwei Kugeln. Trifft die nächste Kugel darauf, bleibt sie wieder fest kleben, und so geht es immer weiter. Auf diese Weise entsteht um die Urkugel eine immer größere Figur, an welcher immer mehr Kugeln kleben bleiben. Man könnte erwarten, dass ein hässlicher, unförmiger Klumpen entsteht. Tatsächlich ist dies nicht der Fall: Es entsteht etwas Interessantes und Schönes.

Dies ist aber erst bei sehr vielen Kugeln zu erkennen. Das folgende Bild zeigt, wie sich die ersten neun Kugeln aneinander kleben könnten:



### Aufgabe

1. Schreibe ein Programm, das dieses Experiment mit möglichst vielen Kugeln durchführen und das entstandene Bild ausgeben kann. Diskutiere dazu verschiedene Möglichkeiten, den Zufallsprozess (unter Benutzung eines vorhandenen Zufallszahlengenerators) zu modellieren. Implementiere eine Variante und begründe deine Wahl.
2. Erstelle ein Bild mit etwa 100 Kugeln und mindestens zwei weitere Bilder mit sehr vielen mehr.

## Aufgabe 2: Spinnen im Netz

Der Cybermeister GOTE SAN sendet seinen Schülern geheime Botschaften. Er gibt ihnen dazu Programme, die er Cyberspinnen nennt, welche die Verzeigerungen auf GOTEs Website analysieren. Cyberspinnen suchen, ausgehend von einer bestimmten Startseite, bestimmte Unterstrukturen der Verzeigerungen zwischen den einzelnen HTML-Seiten.

GOTE ändert seine Website täglich. Findet eine Cyberspinne eine bestimmte Unterstruktur, dann signalisiert dies dem Schüler eine vorher verabredete Botschaft seines Meisters.

Die Cyberspinne der Schülerin THORA TRYPER sucht zum Beispiel nach einer Struktur, die einem Tetraeder entspricht. Sie wird fündig, wenn GOTE (zum Beispiel) die folgenden Dateien auf seiner Website innerhalb eines Verzeichnisses ablegt:

<pre>a.html: ... ...href="b.html" ... ...href="c.html" ... ... c.html: ... ...href="d.html" ... ...</pre>	<pre>b.html: ... ...href="c.html" ... ...href="d.html" ... ... d.html: ... ...href="a.html" ... ...href="b.html" ... ...</pre>
---	--

Betrachtet ein Außenstehender das Treiben, kann er die Botschaften nicht entschlüsseln. Er kann noch nicht einmal feststellen, wann eine Botschaft gesendet wird.

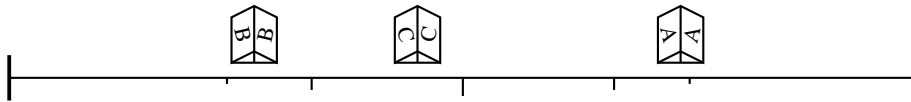
### Aufgabe

1. Überlege dir ein Format zur Beschreibung der (Unter-)Strukturen.
2. Schreibe ein Programm „Cyberspinne-S“, das eine beliebige in diesem Format beschriebene Unterstruktur in der Verzeigerungsstruktur einer beliebigen Website erkennen kann. Gehe dabei von einfachen HTML-Seiten (ohne Kommentare, Frames, Java-Script etc.) aus.
3. Organisiere eine besonders geheimniswahrende Weise, wie GOTE SAN Unterstrukturbeschreibungen und die assoziierten Bedeutungen an seine Schüler senden kann. Begründe, warum so das Geheimnis besonders gut gewahrt bleibt.
4. Schreibe ein Programm „Cyberspinne-M“, das eine Website mit  $n$  HTML-Pages erzeugt, deren Struktur eine bestimmte Unterstruktur von  $m < n$  HTML-Pages enthält. Die sonstigen Inhalte der HTML-Pages sind nicht relevant.
5. Benutze Beispiele für Unterstrukturen mit interessanten Entsprechungen, die auch den Ansprüchen eines Cybermeisters genügen.

## Aufgabe 3: Strand

Der ein Kilometer lange aber schmale Strand von Norderfrierum wird im Sommer von zeltenden Rentnern bevölkert, die über mehrere Monate verteilt einzeln oder familienweise anreisen, aber alle gemeinsam abreisen.

An einem kalten Frühlingstag ist Herr A noch ganz alleine da. Er macht sich deswegen den gesamten Strand untertänig und schlägt sein Zelt irgendwo auf. In der folgenden Woche reist Frau B an, die ihr Zelt in der Strandhälfte aufschlägt, in der nicht schon das Zelt des Herrn A steht, und fortan diese Hälfte ihr eigen nennt, während sich Herr A mit der anderen Hälfte begnügt. Die alten Hasen A und B haben schon vorgesorgt, so dass sich ihre Zelte nicht im selben Stranndrittel befinden, denn dann würde es bei der Anreise des Ehepaars C Probleme geben. Die Eheleute C müssen aber auch aufpassen, denn sonst könnten sie sich zum Beispiel – das Bild zeigt es – zwar im leeren Stranndrittel, aber im selben Strandviertel wie Frau B niederlassen, was bei der Anreise des Herrn D fatal wäre: sie müssten ihr Zelt woanders neu aufstellen.



Am Anfang der Saison verläuft alles friedlich; aber später passiert es immer wieder, dass irgendwelche Fremden aus dem Süden – so behaupten es zumindest böse Zungen – das System nicht kapierten, so dass zum Beispiel Frau G ihr Zelt versetzen muss, wenn der Strand bei der Anreise der Familie L in Zwölfstel eingeteilt wird. Deswegen soll aus der Kurtaxe ein Softwareprogramm bezahlt werden, dem sämtliche Zeltpositionen mitgeteilt werden und das neu Ankommenden einen Vorschlag für die Wahl ihres Zeltplatzes macht (allerdings weiß jeder, dass die meisten Stammgäste nicht im Traum daran denken, das Programm zu befragen).

### Aufgabe

1. Diskutiere die Anforderungen an dieses Programm und seine wünschenswerten Eigenschaften. Es muss eisern an dem Prinzip festgehalten werden, dass der Strand beim Eintreffen des  $n$ ten Besuchers in  $n$  gleich große Streifen eingeteilt wird, auf denen jeweils genau ein Zelt steht, für  $n = 1, 2, \dots$  (zur Not müssen einige wenige Zelte versetzt werden, was aber gar nicht gern gesehen wird).
2. Beschreibe einen geeigneten Algorithmus und implementiere ihn in einem Programm, das die von dir genannten Anforderungen erfüllt.
3. Zeige die Ausgabe deines Programms beim Eintreffen der ersten 20 Strandbesucher, wenn
  - a) jeder Neuankömmling das Programm befragt und seinen Vorschlag befolgt;
  - b) der  $n$ te Neuankömmling den Vorschlag des Programms befolgt, falls  $n$  gerade ist, sich aber ohne Beachtung des Programms 7 Meter östlich der Westgrenze des westlichsten freien Strand- $n$ telns niederlässt, falls  $n$  ungerade ist, für  $n = 1, \dots, 20$ .

Gib in beiden Fällen an, wieviele Versetzungen von bereits aufgestellten Zelten insgesamt vorkommen und wann die erste Zeltversetzung stattfindet.

4. Erläutere die Probleme, die in Verbindung mit Zeltversetzungen auftreten, und schlage Lösungen vor.