



22. Bundeswettbewerb Informatik 2003/2004 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 nicht zulässig. **Einsendeschluss ist der 19. April 2004**, wobei das Datum des Poststempels gilt. Bitte adressieren Sie Ihre Einsendungen wieder an den

Bundeswettbewerb Informatik, Ahrstraße 45, 53175 Bonn

Es gibt drei Aufgaben. **Wichtig und neu:** 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 Mensch-Computer-Interaktion oder der grafischen Ausgabe. 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 zusätzlich Programmablaufprotokolle (kommentierte Probeläufe des Programms); senden Sie uns 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 einzelnen, 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 Aufgabe 1, (c) Unterlagen zur Aufgabe 2 und (d) Unterlagen zur Aufgabe 3.

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 7.1., 4.2., 9.3. und 9.4.2004 oder unter 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). In der newsgroup 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 5.-8.10.2004 in München von BMW und TU München ausgerichtet 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.

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 vierten Mal schreibt der Bundeswettbewerb Informatik gemeinsam mit dem Fachbereich Mensch-Computer-Interaktion 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 2004“, die vom 6.-8.9.2004 in Paderborn 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. Bei den angewandten Kriterien handelt es sich also um

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 Programmfunktionalität auf überraschende Weise nutzbar gemacht wird;

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

Aufgabe 1: Überdeckungsspiel

Radio Gul soll in Lililand senden. Lililand ist quadratisch, mit einer Seitenlänge von 1024 Lilimeter, und vollkommen eben. Gesendet werden soll mit mehreren Antennen mit einheitlicher Reichweite R (in Lilimetern), deren Standorte vorgegeben sind. Den Antennen sollen mit möglichst geringem Kostenaufwand Sendefrequenzen so zugewiesen werden, dass in einem vorgegebenen Anteil der Fläche Lililands Radio Gul störungsfrei empfangen werden kann. Dabei ist Folgendes zu beachten:

Sendefrequenzen: stehen im Abstand von je 1 GulHertz zur Verfügung.

Interferenzen: Wenn Antennen mit geringem Abstand auf der gleichen Frequenz senden, kommt es zu Interferenzen. Um Interferenzen zu vermeiden, sind bei der Frequenzzuweisung folgende Regeln einzuhalten: Sind sich zwei Antennen näher als R Lilimeter, müssen die Frequenzen um mindestens 3 GulHertz differieren; sind sie mindestens R , aber weniger als $2R$ Lilimeter voneinander entfernt, muss die Differenz mindestens 1 GulHertz sein. Verletzt die Frequenzzuweisung diese Bedingungen für ein Antennenpaar, kann eine der beiden Antennen nicht betrieben werden.

Sendefrequenzkosten: Die Nutzung einer Frequenz muss teuer bezahlt werden: Die jährlichen Sendefrequenzkosten betragen 1 TEUR¹ für die erste Frequenz (Basisfrequenz) und $(F+1)$ TEUR für eine Frequenz, die zur Basisfrequenz um F GulHertz differiert.

Aus Gründen der Wirtschaftlichkeit sollten diese Kosten minimiert werden. Da auch der Betrieb der Antennen etwas kostet, ist bei gleichen Sendefrequenzkosten eine Lösung mit weniger Antennen natürlich vorzuziehen.

Überdeckung: Eine Konfiguration aus Antennen und Frequenzen ist eine „ausreichende Überdeckung“, wenn auf mindestens K Prozent der Fläche Lililands Radio Gul empfangen werden kann.

Um die Überdeckung zu berechnen, wird über Lililand ein Raster mit einer Auflösung von 1 Lilimeter gelegt. Die Antennen stehen jeweils in der Mitte eines Rasterquadrates. Alle Rasterquadrate, die vollständig innerhalb des Kreises mit Radius R um die Antenne liegen, tragen zur Abdeckung bei (vgl. Abb. 1). Liegt ein Rasterquadrat nur teilweise im Kreis, so gilt es als nicht abgedeckt (beachte, dass trotzdem Interferenzen in diesen Randquadraten auftreten können).

Aufgabe

1. Programmiere eine einfache grafische Ausgabe von Lililand, die Antennenstandorte, Reichweiten, Frequenzbelegungen und Interferenzen zeigt.
2. Programmiere einen Überdeckungskonfigurator, der für unterschiedliche Vorgaben die entsprechenden Fragen beantworten kann:

¹TEUR steht für tausend Euro, wie früher TDM für tausend DM; sollte Ihnen das zu wenig sein, können Sie T auch als „Teilnehmerfaktor“ lesen und sich Ihre ganz persönlichen Vorstellungen von solchen Kosten machen.

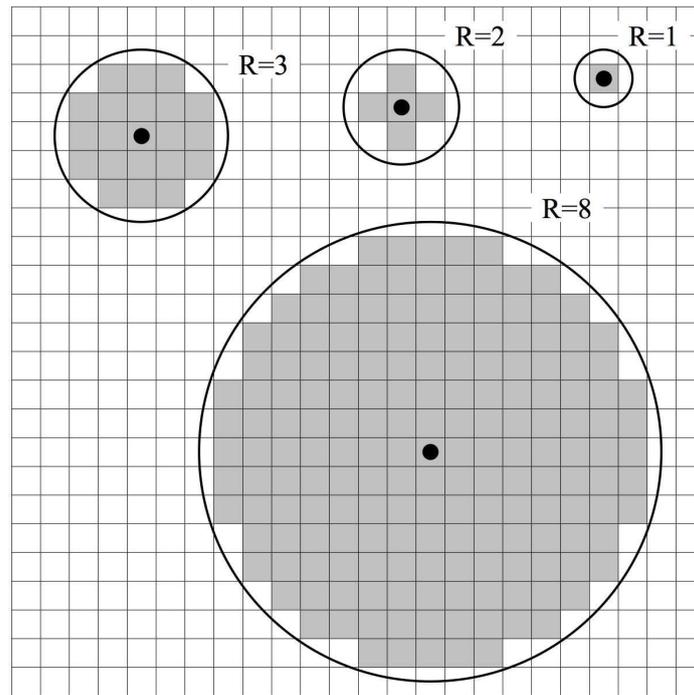


Abbildung 1: Überdeckungen im Raster

- a) Wenn die Antennenstandorte, die Reichweite R und die geforderte Überdeckungs-
zahl K gegeben sind, wie hoch sind dann mindestens die Sendefrequenzkosten für
eine ausreichende Überdeckung?
 - b) Welche Überdeckung kann bei gegebenen Antennenstandorten maximal erreicht
werden, wenn ein Etat von E TEUR verfügbar ist?
3. Bestimme weitere sinnvolle Fragen und erweitere den Konfigurator so, dass er auch
diese Fragen beantworten kann.

Aufgabe 2: Handlungsreisegruppe

Jeder weiß, dass der Handlungsreisende eine schwierige Aufgabe vor sich hat, wenn er plant, in welcher Reihenfolge er die Städte seines Bezirks abklappern soll, um insgesamt möglichst wenig zu reisen. Deswegen bekommt er jetzt zwei Kolleginnen zur Seite gestellt, die ihm einen Teil der Arbeit abnehmen.

Gegeben ist also eine Anzahl Städte in einer ebenen Region. Diese sind in drei Gruppen einzuteilen, und für jede Gruppe ist eine Rundreise festzulegen, die an einer Stadt der Gruppe anfängt, danach alle anderen Städte der Gruppe nacheinander besucht und schließlich zur ursprünglichen Stadt zurückkehrt. Die Einteilung der Städte in drei disjunkte Gruppen darf beliebig sein.

Aufgabe

1. Schreibe ein Programm, mit dem man Städte in der Ebene sowie Reiserouten gut visualisieren kann.
2. Schlage mindestens zwei verschiedene sinnvolle Parameter vor, anhand derer man die Qualität einer Drei-Rundreisen-Lösung definieren kann (in dem herkömmlichen Fall von nur einer Rundreise benutzt man häufig ihre Gesamtlänge). Wähle einen der Parameter und erweitere dein Programm so, dass es den Wert des Parameters bei einer vorliegenden Drei-Rundreisen-Lösung berechnet.
3. Schreibe ein Programm, das zu einer vorgegebenen Menge von Städten eine möglichst gute Drei-Rundreisen-Lösung (nach dem in Teil 2 gewählten Kriterium) berechnet. Das Programm soll auch eine möglichst gute Aussage darüber machen, wie weit die berechnete Lösung von einer optimalen Lösung abweicht (z.B.: „Diese Lösung hat den XYZ-Wert 263, und das ist höchstens 31% mehr als der kleinste mögliche XYZ-Wert“).

Aufgabe 3: Synonymie

(*synonym = gleichbedeutend, bedeutungsähnlich, bedeutungsverwandt, sinngleich, sinnähnlich, sinnverwandt*)

1. In einem Synonymwörterbuch können Wörter paarweise als synonym gekennzeichnet werden. Alle Wörter werden durch eine für alle Wörter gleiche Menge von Attributen beschrieben.

Lege ein Synonymwörterbuch an, bei dem jedes Wort zunächst nur das Attribut „Schriftform“ vom Typ „Zeichenkette“ hat.

Definiere und implementiere Operationen, mit denen die Menge der enthaltenen Wörter, die Menge der die Wörter charakterisierenden Attribute und die Synonymbeziehungen zwischen den Wörtern manipuliert werden können.

2. Eine einfache Benutzungsoberfläche für das Synonymwörterbuch und seine Operationen dient dazu, es auszubauen, zu verwalten und sich einzelne Wörter mit ihren Attributen und Synonymen anzuschauen.

Wie muss die Benutzungsoberfläche gestaltet werden, damit sie eine effektive Arbeit mit dem Wörterbuch ermöglicht, insbesondere beim Ausbau des Wortbestandes? Gib entsprechende Kriterien an.

Implementiere eine solche einfache Benutzungsoberfläche. Erstelle damit ein konkretes Synonymwörterbuch von 100 Wörtern mit fünf Attributen.

3. Gibt man einem „Synonymisator“ eine Wortfolge ein, dann ersetzt er deren Wörter nach bestimmten Regeln durch Synonyme, unter Verwendung eines Synonymwörterbuchs. Es gibt verschiedene, auswählbare Regeln. Sie beziehen sich auf Attribute der Wörter, die entweder explizit eingeführt wurden (z. B. „Grad der Förmlichkeit“) oder die automatisch ermittelt werden können (z. B. „Wortlänge“).

Programmiere einen Synonymisator. Erfinde originelle Regeln, programmiere sie und erzeuge mit Hilfe des erstellten Wörterbuchs für jede Regel mindestens zwei Synonymisationsbeispiele.

Ein Beispiel für eine Regel: „Ersetze jedes Wort durch ein Synonym, dessen Schriftform den Buchstaben t nicht enthält.“