

# Welfenlab Competition 2007

## Schülerwettbewerb Informatik

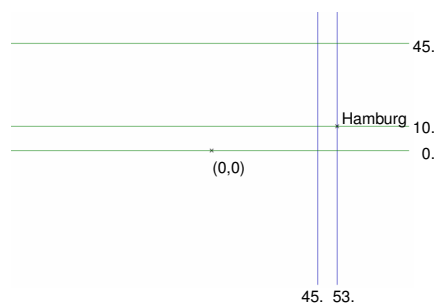
### Aufgabenblatt 2

<http://www.welfenlab.de/competition/>

In dieser Aufgabe sollen Koordinaten auf der Erde in den aus Aufgabenblatt 1 bekannten Koordinatendarstellungen graphisch dargestellt werden. Dabei sollen sowohl Städte als Punkte auf einer Ebene, als auch die Längen- und Breitengrade in die Ausgabe grafik abgebildet werden. Außerdem sollen der kürzeste Weg zwischen zwei Städten gefunden und graphisch dargestellt werden. Im Folgenden ist erklärt, wie die einzelnen Koordinatendarstellungen auf die Bildebene gezeichnet werden sollen.

#### Darstellungsart 1: Kugelkoordinaten

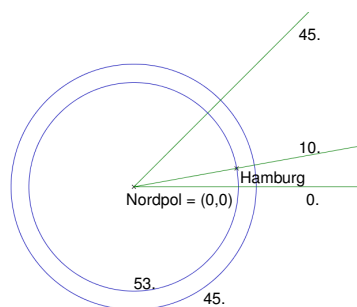
Für die graphische Darstellung in Kugelkoordinaten wird  $\Phi$  auf die y-Achse aufgetragen und  $\Theta$  auf die x-Achse des Bildes. Somit beträgt der darzustellende Wertebereich  $-90^\circ$  bis  $+90^\circ$  in x-Richtung und  $-180^\circ$  bis  $+180^\circ$  in y-Richtung.



Kugelkoordinatendarstellung des 45. und des 53. Breitengrades, des 0., 10. und 45. Längengrades, sowie der Stadt Hamburg und des Schnittpunktes von Nullmeridian und Äquator.

#### Darstellungsart 2: Orthographische Koordinaten

Für die graphische Ausgabe von orthographischen Koordinaten werden Nord- und Südhalbkugel getrennt dargestellt. Für die Darstellung der Halbkugeln werden jeweils die  $xy$ -Koordinaten des Punktes in orthographischer Darstellung verwendet, also  $x_o$  und  $y_o$ . Der darzustellende Wertebereich beträgt also sowohl in x-Richtung als auch in y-Richtung  $-1$  bis  $+1$ .



Orthographische Darstellung des 45. und des 53. Breitengrades, des 0., 10. und 45. Längengrades, sowie der Stadt Hamburg und des Nordpols.

### Darstellungsart 3: Stereographische Koordinaten

Für stereographische Koordinaten werden zwei verschiedene graphische Darstellungen betrachtet. Diese repräsentieren unterschiedliche Methoden den unendlichen Originalwertebereich in einen endlichen Bereich zu bringen:

**Eingeschränkte Stereographische Koordinaten:** Für diese Darstellung wird ein minimaler Breitengrad mit angegeben. In der graphischen Darstellung werden nur Koordinaten dargestellt, die über diesem Breitengrad liegen (also deren Entfernung zum Südpol größer ist als die des minimalen Breitengrades). Die Positionierung auf der Bildebene geschieht dann wiederum über die direkte Verwendung der  $xy$ -Koordinaten des Punktes in stereographischer Darstellung. Der Wertebereich für die Darstellung ist abhängig vom minimalen Breitengrad zu wählen, das heißt die minimalen und maximalen  $x$ - und  $y$ -Werte werden aus den minimalen und maximalen  $x$ - und  $y$ -Werten des minimalen Breitengrades berechnet.

**Transformierte Stereographische Koordinaten:** Hier werden die  $xy$ -Koordinaten der stereographischen Darstellung transformiert um sie in einen Wertebereich von  $-1$  bis  $+1$  zu überführen. Im Folgenden seien  $x_B$  und  $y_B$  die Koordinaten für die graphische Ausgabe. Dann lautet die Transformationsformel:

$$\begin{aligned}x_B &= 2 \cdot \arctan(x_S)/\pi \\y_B &= 2 \cdot \arctan(y_S)/\pi\end{aligned}$$

Somit entspricht der darzustellende Wertebereich wiederum  $-1$  bis  $+1$  in beiden Richtungen.

### Darstellungsart 4: Gnomonische Koordinaten

Für gnomonische Koordinaten werden wie für stereographische Koordinaten zwei verschiedene graphische Darstellungen betrachtet:

**Eingeschränkte Gnomonische Koordinaten:** Wie für orthographische Koordinaten werden Nord- und Südhalbkugel getrennt dargestellt. Ansonsten entspricht die Darstellung im Wesentlichen der für eingeschränkte stereographische Koordinaten, das heißt, dass ein minimaler Breitengrad  $\Phi_0$  angegeben wird und nur Koordinaten, die über  $\Phi_0$  (Nordhalbkugel) bzw. unter  $-\Phi_0$  (Südhalbkugel) liegen, dargestellt werden. Der minimale Breitengrad muss für diese Darstellung über dem Äquator liegen. Die Positionierung auf der Bildebene geschieht dann wiederum über die direkte Verwendung der  $xy$ -Koordinaten des Punktes in gnomonischer Darstellung.

**Transformierte Gnomonische Koordinaten:** Wiederum werden Nord- und Südhalbkugel getrennt dargestellt. Außerdem werden die  $xy$ -Koordinaten der gnomonischen Darstellung mit der Transformationsformel für die transformierte stereographische Darstellung transformiert und damit in den Wertebereich  $-1$  bis  $+1$  überführt.

## Dateiformat

Das aus dem ersten Aufgabenblatt bekannte Dateiformat wird für diese Aufgabe erweitert. Es besteht nun aus drei Segmenten, die jeweils durch eine Zeile, welche lediglich eine # enthält voneinander getrennt sind.

Im ersten Segment befindet sich eine Liste von Städten in der aus dem ersten Aufgabenblatt bekannten Formatierung.

Im zweiten Segment befindet sich eine Liste zu zeichnender Längen- und Breitengrade. Dabei wird jeweils ein Längen- bzw. Breitengrad pro Zeile gespeichert. Die erste Angabe in einer Zeile gibt an, ob es sich um einen Breitengrad (B) oder einen Längengrad (L) handelt. Danach folgt durch ein Semikolon getrennt die Winkelangabe für diesen Längen- bzw. Breitengrad in Grad.

Im dritten Segment befindet sich eine Liste von Städtepaaren, für die die kürzesten Wege zu berechnen und darzustellen sind. Dafür werden in jeder Zeile die Namen der Start- und der Zielstadt durch ein Semikolon getrennt angegeben.

Für die Koordinatenangaben reicht wieder eine Genauigkeit von 4 Stellen.

```
K;52.5167;13.4167;Berlin
K;51.5;0.1333;London
O;0;0.5848;0.1035;Hamburg
S;0.6646;0.114;Hannover
G;1;-1.306;0.7174;Sydney
#
B;10.0
L;20
B;-25.1
B;65.3465
L;40.2375
L;-40
#
Hamburg;Hannover
Sydney;London
Sydney;Hannover
```

## Skalarprodukt

Für die Berechnung der kürzesten Wege auf Kugeln ist das Skalarprodukt (auch inneres Produkt genannt) hilfreich. Das Skalarprodukt eignet sich zur Berechnung des Winkels zwischen zwei Vektoren und ist definiert durch:

$$\vec{a} \cdot \vec{b} := \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} \cdot \begin{pmatrix} b_x \\ b_y \\ b_z \end{pmatrix} := a_x \cdot b_x + a_y \cdot b_y + a_z \cdot b_z$$

Für das Skalarprodukt gilt, wenn  $\alpha$  der Winkel zwischen den Vektoren  $\vec{a}$  und  $\vec{b}$  ist:

$$\cos(\alpha) = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|} \text{ mit } \|\vec{a}\| := \sqrt{a_x^2 + a_y^2 + a_z^2}$$

Als kleiner Tip sei hier noch erwähnt, dass der Betrag eines Winkels im Bogenmaß der Länge des durch ihn beschriebenen Kreisbogens auf dem Einheitskreis entspricht.

## Aufgabenstellung

Es soll ein Programm zur graphischen Darstellung der verschiedenen Koordinatendarstellungen, sowie der Berechnung und Darstellung der kürzesten Wege zwischen zwei Punkten auf einer Kugel erstellt werden. Folgende Teilaufgaben sollen gelöst werden:

1. Erstelle ein Programm, das eine Liste von Städten und Längen- und Breitengraden aus einer Datei einliest und diese in einer einstellbaren Koordinatendarstellung graphisch ausgibt. Die sechs genannten graphischen Darstellungen (Darstellung von Kugelkoordinaten, orthographischen Koordinaten, eingeschränkten stereographischen Koordinaten, transformierten stereographischen Koordinaten, eingeschränkten gnomonischen Koordinaten, transformierten gnomonischen Koordinaten) müssen alle wählbar sein. Für die eingeschränkten Darstellungen muss zudem der minimale Breitengrad einstellbar sein (durch Kommandozeile oder eine GUI). Bei der Darstellung von gnomonischen und orthographischen Koordinaten müssen jeweils beide Halbkugeln klar getrennt gezeichnet werden. Die eigentliche Darstellung kann entweder im Programm intern oder als Ausgabe in eine Bitmap geschehen. Das Abspeichern einer Bitmap ist beispielsweise mit `sdl_image` möglich. Bei der Darstellung sollten Städte in schwarz oder weiß, Breitengrade in blau und Längengrade in grün gezeichnet werden. Für das Zeichnen von Verbindungen oder z.B. Längengraden könnt ihr eine ausreichend große Menge von Punkten auf der zu zeichnenden Kurve berechnen und diese durch Geradenstücke verbinden (achtet dabei darauf, dass die Kurve nicht zu eckig wird).
2. Erweitere das Programm dahingehend, dass es die kürzesten Wege für die in der Eingabedatei angegebenen Verbindungen berechnet, die Weglänge in Kilometern ausgibt (gehe dazu von einem Erdradius von 6371 Kilometern aus) und die Verbindungsstrecken mit in die graphische Ausgabe zeichnet (diese sollten in rot eingezeichnet werden). Auch die Verbindungsstrecken dürfen über ausreichend viele Punkte angenähert werden.
3. Erweitere die Städteliste aus Aufgabenblatt 1 um je 4 Längengrade und Breitengrade und 3 Verbindungsstrecken zwischen Städten. Verwende diese Datei als Eingabedaten.
4. In der Ausarbeitung müssen alle benutzten Algorithmen und der Programmaufbau verständlich dargestellt werden. Insbesondere: Warum liefert den Algorithmus tatsächlich den kürzesten Weg.

Beachte, dass die Berechnung der kürzesten Verbindungsstrecke wirklich nicht einfach ist. Man kann z.B. den Startvektor durch Drehung in den Zielvektor überführen. Solltet ihr Probleme damit haben meldet euch bitte per Email.

## Teilnahmebedingungen

- Eine nachträgliche Anmeldung ist weiterhin möglich. Das erste Aufgabenblatt wird jedoch in diesem Fall nicht mehr bewertet. Trotzdem sind verspätete Teilnehmer selbstverständlich willkommen und werden auch bewertet.
- Das erstellte Programm und die Ausarbeitung sind bis zum 16.12.2007 bei uns einzureichen. Die nächste Aufgabe wird am 17.12.2007 veröffentlicht.
- Als Programmiersprache sind C, C++, Pascal/Delphi und Java zugelassen. Es dürfen nur die Standard-Bibliotheken und frei verfügbare GUI-Frameworks wie z.B. wxWindows, Qt oder Gtk verwendet werden.
- Es muss pro Aufgabenblatt eine zwei- bis vierseitige Ausarbeitung angefertigt werden, in der die benutzten Algorithmen erklärt werden und in der das Programm ausführlich dokumentiert wird. Außerdem müssen alle Quellen und Hilfsmittel angegeben werden.

- Es dürfen nur Schülerinnen und Schüler der Sekundarstufe I oder II allgemeinbildender Schulen aus Niedersachsen an dem Wettbewerb teilnehmen. Familienangehörige von Mitarbeitern des Fachgebietes Graphische Datenverarbeitung an der Universität Hannover sind leider ausgeschlossen.
- Der Rechtsweg ist ausgeschlossen.

**Bewertungskriterien**

- Lauffähigkeit und Korrektheit des Programms
- Gut verständliche Dokumentation der Algorithmen
- Besondere eigenständige Ideen
- In Zweifelsfällen: Strukturierter Programmierstil
- ...

Bei Problemen oder Rückfragen melde dich bitte bei uns.  
E-Mail: [competition@welfenlab.de](mailto:competition@welfenlab.de)

Viel Erfolg wünschen  
Prof. Dr. F.-E. Wolter und das Team der Welfenlab Competition!