

Entwicklung eines Softwaresystems

Hintergrund der Aufgabe

Ihr Bekannter Max Matse wendet sich hilfesuchend an Sie, da er weiß, dass Sie Programmierer sind. Er hat ein Online-Strategie-Spiel entdeckt, bei dem er eine Armee delegieren muss, also u.a. entscheidet, welchen Weg die Armee von einem Start- zu einem Endpunkt zurücklegen soll. Das Land wurde bereits kartographiert, d.h. es wurde in gleichgroße, rechteckige Elemente in vertikaler und horizontaler Richtung zerlegt. Jedes Element ist durch einen Index (x,y) eindeutig identifizierbar: x repräsentiert die horizontale Position, y die vertikale. Das allererste Element oben links bekommt den Index (1,1), das letzte Element ganz unten rechts bekommt den Index (20,20).

Dabei ist nicht jeder Weg zwischen Start- und Zielpunkt als gleichwertig anzusehen, denn das zu überquerende Terrain ist unterschiedlich beschaffen. Die verschiedenen Terrainarten sind verschieden schwer zu überqueren: Für die Überquerung von schwierigem Terrain wie Bergketten und Flussgebiete benötigt man länger als wie von Waldgebieten und weiten Ebenen. Max möchte den Weg ermitteln, für den seine Armee am wenigsten Zeit benötigt und nicht jedes Mal selber im Kopf die unterschiedlichen Wegszenarien durchspielen, sondern er bittet Sie um ein Programm, dass das Problem löst.

Bestimmung eines zeitminimalen Weges

Max formuliert also folgende Aufgabenstellung an Sie:

Ein rechteckiges, sehr terrain-abwechslungsreiches Gebiet (Berge, Wüste, Sumpf, ...) liegt als Raster mit m x n Elementen vor. Hierbei sind maximal 20 x 20 Elemente möglich. Es soll nun die Dauer verschiedener Wege durch das Gebiet berechnet werden (Bsp.: Das Startelement befindet sich im Element an Stelle (2,1), der Zielpunkt bei (12, 14)). Die mittlere Dauer für die Durchquerung eines Gebietsstücks ist für jedes Element bekannt – dabei werden ganzzahlige Werte zwischen 1 und 10 verwendet, wobei 1 bedeutet, dass sich das Gebiet sehr schnell durchqueren lässt und 10, dass die Durchquerung sehr schwer und zeitintensiv ist. Es soll hierfür ein Weg mit minimierten Kosten gefunden werden. Dafür muss zunächst ein möglicher Weg mit der zugehörigen Zeit bestimmt werden als Referenzweg. Anschließend erfolgt die Ermittlung der zeitminimalen Lösung.

Für die Verbindungen gilt, dass sie nur vertikal oder horizontal sein dürfen, diagonal ist nicht möglich. Existieren mehrere Wege mit gleicher minimaler Zeit, soll der Weg genommen werden, der die wenigsten Elemente beinhaltet.

Ein- und Ausgabe

Für das Eingabeformat fordert Max:

- 1. Zeile: Kommentarzeile, die das Gebiet beschreibt. Kommentare beginnen mit „%“.
- Zeitangaben für die Gebiets-Elemente. Die Zeitangaben werden durch Kommata voneinander getrennt. Die Größe des Gebiets erhält man durch die Anzahl der Elemente der ersten Elementzeile und durch die Anzahl der eingelesenen Zeilen.
- Start- und Zielelement beinhalten keine Zeitangabe, sondern ein „S“ bzw. ein „Z“.

Für das Ausgabeformat wünscht sich Max folgendes:

- 1. Zeile: Der Text des Kommentars aus der Eingabedatei findet sich auch hier wieder
- 2. Zeile: Anzahl der Elemente in vertikaler und horizontaler Richtung
- 3. Zeile: Index des Startelements
- 4. Zeile: Index des Endelements
- 5. Zeile: Initiale Abschätzung der Zeitobergrenze
- 6. Zeile: Berechnete minimale Zeit
- ab der 7. Zeile: Der von S zu Z führende Weg

Beispiel 1 mit 7x7 Elementen:

Eingabedatei:

%Waldgebiet mit 7 x 7 Elementen

```

10, 10, 10, 10, 10, 10, 10
10, S, 10, 10, 10, 10, 10
10, 2, 1, 10, 10, 10, 10
10, 10, 1, 10, 10, 10, 10
10, 2, 1, 2, 1, Z, 10
10, 10, 10, 10, 10, 10, 10
10, 10, 10, 10, 10, 10, 10

```

Die Datei ergibt folgendes Gebiet:

10	10	10	10	10	10	10
10	S	10	10	10	10	10
10	2	1	10	10	10	10
10	10	1	10	10	10	10
10	2	1	2	1	Z	10
10	10	10	10	10	10	10
10	10	10	10	10	10	10



bei dem grauen Weg handelt es sich um die minimale Lösung, der Weg durch den die Linie führt ist der Referenzweg zur Bildung der Zeitobergrenze.

Ausgabedatei:

Waldgebiet mit 7x 7 Zeilen

Startelement: (2/2)

Zielelement: (5/6)

Abschätzung der Zeitobergrenze: 18

Minimalzeit: 8

Weg: S, (3/2), (3/3), (4/3), (5/3), (5/4), (5/5), Z

Beispiel 2 mit 7x7 Elementen:

Eingabedatei:

%Sumpfgebiet mit 7x 7 Elementen

```

10, 10, 10, 10, 10, 10, 10
10, S, 2, 2, 10, 10, 10
10, 1, 1, 2, 2, 2, 10
10, 10, 1, 10, 10, 1, 10
10, 2, 1, 2, 9, Z, 10
10, 10, 10, 10, 10, 10, 10
10, 10, 10, 10, 10, 10, 10

```

Die Datei ergibt folgendes Gebiet:

10	10	10	10	10	10	10
10	S	2	2	10	10	10
10	1	10	2	2	2	10
10	10	1	10	10	1	10
10	2	1	2	9	Z	10
10	10	10	10	10	10	10
10	10	10	10	10	10	10

bei dem grauen Weg handelt es sich um die minimale Lösung, der Weg durch den die Linie führt ist der Referenzweg zur Bildung der Zeitobergrenze. Der blaue Weg wird nicht berücksichtigt, da man für ihn 15 Zeiteinheiten benötigt.

Ausgabedatei:

Sumpfgebiet mit 7 x 7 Zeilen

Startelement: (2/2)

Zielelement: (5/6)

Abschätzung der Zeitobergrenze: 25

Minimalzeit: 11

Weg: S, (2/3), (2/4), (3/4), (3/5), (3/6), (4/6), Z

Beispiel 3 mit 20 x 3 Elementen:

Eingabedatei:

%Flussgebiet mit 20 x 3 Elementen

S, 1, 1, 10, 10, 10, 1, 1, 1, 10, 10, 1, 1, 1, 10, 10, 1, 1, 1, 10
10, 10, 1, 10, 10, 10, 1, 10, 1, 10, 1, 1, 10, 1, 1, 10, 1, 10, 1, 1
10, 10, 1, 1, 1, 1, 1, 10, 1, 1, 1, 10, 10, 10, 1, 1, 1, 10, 10, Z

Die Datei ergibt folgendes Gebiet:

S	1	1	10	10	10	1	1	1	10	10	1	1	1	10	10	1	1	1	10
10	10	1	10	10	10	1	10	1	10	1	1	10	1	1	10	1	10	1	1
10	10	1	1	1	1	1	10	1	1	1	10	10	10	1	1	1	10	10	Z

bei dem grauen Weg handelt es sich um die minimale Lösung, der Weg durch den die Linie führt ist der Referenzweg zur Bildung der Zeitobergrenze.

Ausgabedatei:

Flussgebiet mit 20 x 3 Zeilen

Startelement: (1/1)

Zielelement: (3/20)

Abschätzung der Zeitobergrenze: 101

Minimalzeit: 32

Weg: S, (1/2), (1/3), (2/3), (3/3), (3/4), (3/5), (3/6), (3/7), (2/7), (1/7),
(1/8), (1/9), (2/9), (3/9), (3/10), (3/11), (2/11), (2/12), (1/12), (1/13),
(1/14), (2/14), (2/15), (3/15), (3/16), (3/17), (2/17), (1/17), (1/18), (1/19),
(2/19), (2/20), Z

Aufgabenstellung

Sie haben Max versprochen, dass von ihm gewünschte Programm zu schreiben, damit er zukünftig schneller (und mit weniger Einsatz von Gehirnschmalz) den zeitgünstigsten Weg

Existieren mehrere günstigste Lösungen, reicht es, eine beliebige dieser günstigsten Lösungen zu bestimmen.

Testen Sie Ihr Programm, diskutieren Sie dabei Ihre Testbeispiele.

Tipp:

- Eine hilfreiche Möglichkeit bei der Suche nach der günstigsten Lösung kann das Suchen einer leicht zu bestimmenden und recht günstigen ersten Lösung als Referenzlösung sein.
- Die grafische Aufarbeitung der Daten ist nicht verlangt.

Im Rahmen der schriftlichen Aufgabe am ersten Tag sind abzugeben:

- Aufgabenanalyse und Entwurf des Algorithmus
 - Aufgabenbeschreibung, d.h. Analyse der Problemstellung sowie Diskussion einer geeigneten Programm- und Datenstruktur für das angegebene Problem
 - Einlesen und Initialisieren der Daten
 - Entwurf des Algorithmus
 - Ausgabe gemäß Aufgabenstellung
- Programmkonzeption unter Berücksichtigung der funktionalen Trennung
 - Klassen, Schnittstellen, Methoden und Datenstrukturen in Form von UML-Diagrammen
 - UML-Sequenzdiagramme für die wesentlichen Abläufe
 - Detaillierte Beschreibung der wesentlichen Methoden in Form von Nassi-Schneidermann-Diagrammen

Im Rahmen des Prüfproduktes sind gedruckt abzugeben:

- Verbale Beschreibung und Diskussion des realisierten Verfahrens
- Programmsystem (Klassen, Schnittstellen, Methoden) als Quellcode und in ausführbarer Form
- Entwicklerdokumentation
- Benutzeranleitung
- ausführliche Beschreibung, Begründung und Diskussion
 - der angegebenen Beispiele sowie
 - eine ausreichende Zahl von Beispielen mit Ein- und Ausgabedateien, die sowohl die Normalfälle als auch auftretende Spezial- und Fehlerfälle abdecken
- Zusammenfassung und Ausblick, z.B. Erweiterungsmöglichkeiten
- Ein- und Ausgabedateien der Beispiele
- ein Skript zur automatischen Ausführung aller Beispiele