

Concours Informatique Luxembourgeois 2001

Epreuve Finale Tranche II

Aufgabe I - Zahlen ordnen

Die N ganzen Zahlen von 1 bis N werden in einer beliebigen Reihenfolge gegeben. In einem Verarbeitungsschritt werden jeweils zwei dieser Zahlen vertauscht. Es gilt herauszufinden, wieviel dieser Arbeitsschritte mindestens benötigt werden um die Zahlen in die normale aufsteigende Reihenfolge zu bringen. Als Antwort gibt man die Anzahl der Tauschvorgänge und dann jeweils die Positionen der zu tauschenden Zahlen an.

Beispiel

1 3 4 2 5 ergibt 1 4 3 2 5 (die zweite und die dritte Zahl tauschen)
und danach 1 2 3 4 5 (die zweite und die vierte Zahl tauschen).

Eingabe

Die Eingabedatei ist die Textdatei ZAHLEN.IN.

Die erste Zeile enthält eine ganze Zahl N ($1 \leq N \leq 1000$).

Die zweite Zeile enthält die N Zahlen, jeweils durch ein Leerzeichen getrennt.

Ausgabe

Die Ausgabedatei ist die Textdatei ZAHLEN.OUT.

Die erste Zeile enthält eine ganze Zahl, die minimale Anzahl der Tauschoperationen.

Die weiteren Zeilen enthalten jeweils 2 Zahlen, die Positionen der zu tauschenden Zahlen, auch jeweils durch ein Leerzeichen getrennt.

Beispiel Ein- und Ausgabe

ZAHLEN.IN	ZAHLEN.OUT
5	2
1 3 4 2 5	2 3
	2 4



Speichert das Programm auf die Diskette unter dem Namen ZAHLEN.xxx (wo xxx=PAS oder XXX=C). Die Diskette muss außerdem die binäre Datei ZAHLEN.EXE enthalten (ausführbares Programm).

Aufgabe II - Der Alpinist

Ein Bergsteiger befindet sich in der nordwestlichen Ecke eines quadratischen, hügeligen Geländes und möchte zur gegenüberliegenden, südöstlichen Ecke wandern. Bewegt er sich topographisch gesehen auf der gleichen Höhe oder steigt er bergab, so hat er beim Atmen keine Schwierigkeiten - doch wenn er eine höhere Position erklimmt, benötigt er zusätzlichen Sauerstoff. Im Bedarfsfall muss er eine Einheit pro Höhenunterschied zu sich nehmen.



Beispiel

Eine Bewegung von einem Punkt der Höhe 7 zu einem Punkt der Höhe 9 erfordert 2 Sauerstoffeinheiten.

Der Alpinist bewegt sich in einzelnen Schritten in eine der vier Himmelsrichtungen (N, O, S, W) - also nie diagonal - und darf die quadratische Region nicht verlassen.

Einschränkungen

- ♦ Da der Alpinist seine gesamte Ausrüstung mittragen muss, sind seine Sauerstoffreserven auf 10 Einheiten beschränkt.
- ♦ Da man im Alpinismus aus biologischen Gründen nicht zu schnell hochsteigen darf, kann der Bergsteiger pro Schritt nie mehr als 3 Höheneinheiten im Aufstieg überwinden.

Die nordwestliche Ecke des Planquadrats hat die Position (1,1), der südöstliche Punkt hat die Koordinaten (n,n), wobei n ebenso wie die Höhe h jedes einzelnen Punktes (x,y) eine positive ganze Zahl ist:

$$1 \leq n \leq 30$$

$$1 \leq x \leq n$$

$$1 \leq y \leq n$$

$$1 \leq h \leq 9$$

Schreibe ein Programm, welches für den Bergsteiger einen Weg findet, auf dem er möglichst wenig Sauerstoff verbraucht.

Eingabe

Die Eingabedatei ist die Textdatei ALPIN.IN.

Die erste Zeile enthält die Größe n des quadratischen Geländes.

Die nächsten n Zeilen enthalten je n Zahlen, jeweils durch ein Leerzeichen getrennt, die den Höhen h der jeweiligen Punkte entsprechen, und zwar in der Reihenfolge

(1,1) (1,2) (1,3) ... (1,n)

(2,1) (2,2) (2,3) ... (2,n)

(3,1) (3,2) (3,3) ... (3,n)

...

(n,1) (n,2) (n,3) ... (n,n)

Ausgabe

Die Ausgabedatei ist die Textdatei ALPIN.OUT. Falls eine Überquerung des Geländes möglich ist, so ist das Ergebnis die Anzahl der minimal aufzunehmenden Sauerstoffeinheiten. Ist es nicht möglich zum Ziel zu gelangen, so enthält die Datei den Wert -1.

Beispiel Ein- und Ausgabe 1

ALPIN.IN	ALPIN.OUT
5	3
5 4 3 2 1	
7 5 6 6 6	
8 8 8 9 7	
9 6 9 9 6	
4 5 4 5 3	

Beispiel Ein- und Ausgabe 2

ALPIN.IN	ALPIN.OUT
3	-1
2 3 9	
2 7 9	
1 8 9	



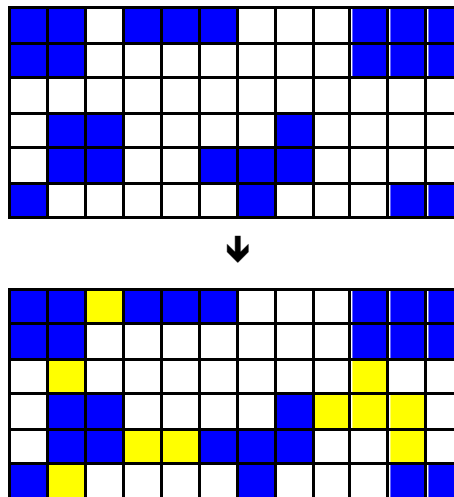
Speichert das Programm auf die Diskette unter dem Namen ALPIN.xxx (wo xxx=PAS oder XXX=C). Die Diskette muss außerdem die binäre Datei ALPIN.EXE enthalten (ausführbares Programm).

Aufgabe III - Kanalisation

Die finnische Seenplatte besteht aus einer gewissen Anzahl von Seen. Es gilt alle Seen in einem Teilstück der Platte mit Kanälen zu verbinden und zwar so dass die Länge aller Kanäle so kurz wie möglich ist. Die Kanalverbindungen dürfen aber nur vertikal oder horizontal sein. Abzweigungen sind jederzeit möglich.



Beispiel



Die minimale Länge aller Kanalverbindungen beträgt für dieses Beispiel 10 Einheiten.

Einschränkungen

Die minimale Länge beziehungsweise Breite des Plattenteilstücks ist 1, der maximale Wert ist 50.

■ Schreibe ein Programm, welches die kürzeste Kanalverbindung zwischen den Seen findet.

Eingabe

Die Eingabedatei ist die Textdatei KANAL.IN.

Die erste Zeile enthält die Länge und die Breite des Geländes, getrennt durch ein Leerzeichen.

Die weiteren Zeilen enthalten die Struktur des Geländes: 0 ≡ kein Wasser; 1 ≡ Wasser. Innerhalb derselben Zeile sind alle Ziffern durch ein Leerzeichen getrennt.

Ausgabe

Die Ausgabedatei ist die Textdatei KANAL.OUT und enthält die minimale Länge des Verbindungskanals. Wenn man kein Kanal braucht um alle Seen miteinander zu verbinden (im Falle einer natürlichen Verbindung der Seen), so enthält die Datei logischerweise den Wert 0.

Beispiel Ein- und Ausgabe

KANAL.IN

12	6										
1	1	0	1	1	1	0	0	0	1	1	1
1	1	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	1	0	0	0	0
0	1	1	0	0	1	1	1	0	0	0	0
1	0	0	0	0	0	1	0	0	0	1	1

KANAL.OUT

10



Speichert das Programm auf die Diskette unter dem Namen KANAL.xxx (wo xxx=PAS oder XXX=C). Die Diskette muss außerdem die binäre Datei KANAL.EXE enthalten (ausführbares Programm).