

Programmierwettbewerb Hackolaus

Regeln

Thomas Liebetraut Fabian Wenzelmann

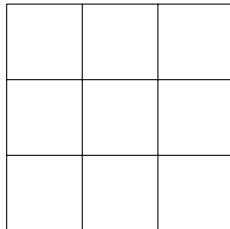
Fachschaft Technische Fakultät der
Albert-Ludwigs-Universität Freiburg

7. Dezember 2012



The Game

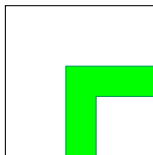
- Wir spielen **Das verrückte Informatikerlabyrinth**.
- Gegeben ist ein $n \times n$ -Spielbrett (für ungerade natürliche Zahlen $n \geq 3$)



- Jedes Spielfeld ist repräsentiert durch eine Karte.

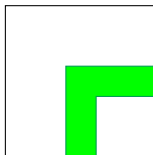
Karten

Jede Karte hat einen aufgedruckten **Weg** auf dem man gehen kann.

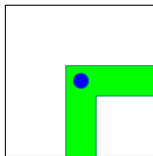


Karten

Jede Karte hat einen aufgedruckten **Weg** auf dem man gehen kann.



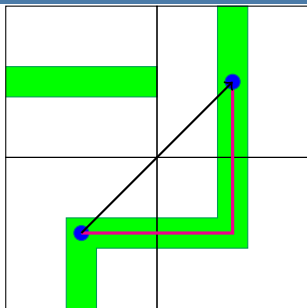
Der Spieler ist durch eine **Spielfigur** auf genau einer Karte repräsentiert.



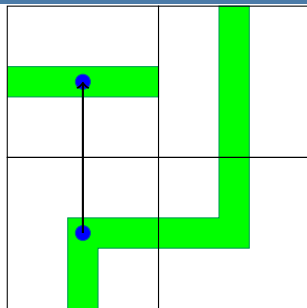
Moving around

Die Spielfigur kann sich nur entlang der aufgedruckten Pfade bewegen.

Example



Gültiger Spielzug



Ungültiger Zug

Informatikerkarten

Es gibt einige Karten, auf denen zusätzlich ein **Informatiker** gedruckt ist.

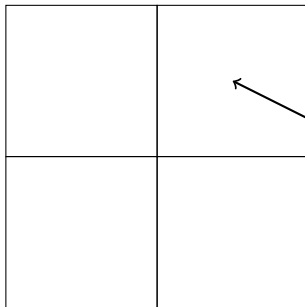


Abbildung: Alan Turing auf einer Karte.

Was ist das Ziel?

- Ihr bekommt ein Spielbrett...
- ...mit der Anordnung der Karten auf dem Brett, ...
- ...welche Informatiker wo sind, ...
- ...und eure Startposition auf dem Spielbrett.

Was ist das Ziel?

- Ihr bekommt ein Spielbrett...
- ...mit der Anordnung der Karten auf dem Brett, ...
- ...welche Informatiker wo sind, ...
- ...und eure Startposition auf dem Spielbrett.
- Gegeben einer Liste von Informatikern, findet einen möglichst kurzen Weg, um alle Informatiker zu besuchen.

Aktionen - Karten verschieben

- Neben den Karten auf dem Spielbrett gibt es eine **zusätzliche Karte**.
- Diese Karte kann Reihen/Spalten **eingeschoben** werden (egal von welcher Seite).
- Dadurch verschieben sich alle Karten der Reihe/Spalte und die Karte auf der gegenüberliegenden Seite wird die freie Karte für den nächsten Zug.
- Aber: Nur jede zweite Reihe/Spalte ist beweglich, also darf nur in Reihen/Spalte mit ungeraden Zahlen (Zählung beginnt bei 0) eingefügt werden.

Aktionen - Karten verschieben

				$(4, 4)$
$(0, 0)$				

Bewegliche Spalten.

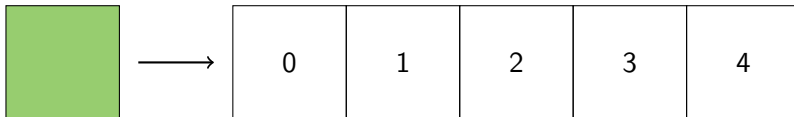
Aktionen - Karten verschieben

				$(4, 4)$
$(0, 0)$				

Bewegliche Reihen.

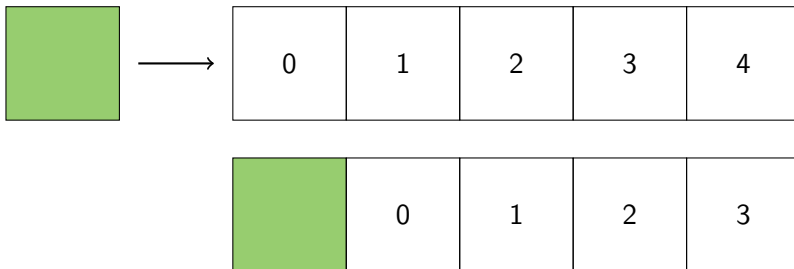
Aktionen - Karten verschieben

Example



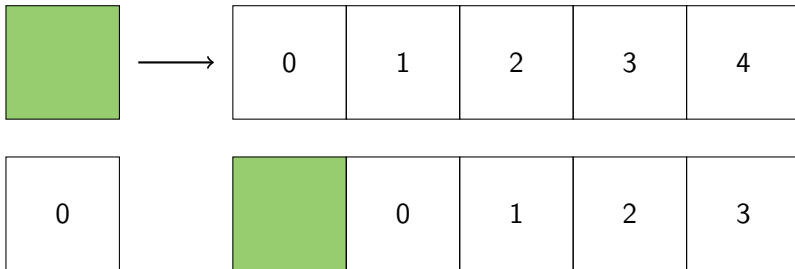
Aktionen - Karten verschieben

Example



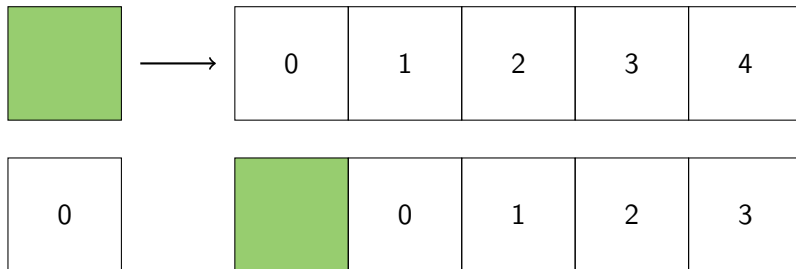
Aktionen - Karten verschieben

Example



Aktionen - Karten verschieben

Example



Note

Schiebt man die Karte mit der Spielfigur heraus, wandert die Spielfigur auf die gegenüberliegende Seite.



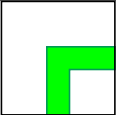
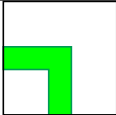
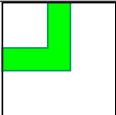
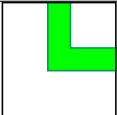

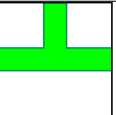
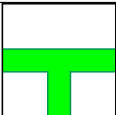
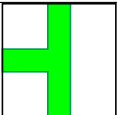
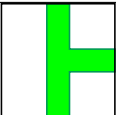
Spielzug

Jeder Spielzug besteht aus zwei Aktionen

1. Eine Reihe oder Spalte verschieben
2. Die Spielfigur auf den resultierenden Wegen beliebig weit verschieben.

Karten

Wir haben die folgenden Kartentypen mit ihren jeweiligen IDs.

			
0	1	2	3
			
4	5	6	7
			
8	9	10	

I/O

- Die Kommunikation findet ausschließlich über Standard-I/O statt.
- Nach dem Starten bekommt ihr eine Problemdefinition über Standard-Input.
- Das Programm löst das Problem.
- Ihr gebt eure Spielzüge im Textformat nach Standard-Output wieder aus.
- Diese Ausgabe wird dann validiert, ob sie auch tatsächlich korrekt ist.

Problemdefinition

1. Die erste Zeile enthält n (= Kantenlänge des Spielbretts)
2. Die zweite Zeile enthält m (= Anzahl der Informatiker, die es zu suchen gilt)
3. Die dritte Zeile enthält die ID der freien Karte, die ihr am Anfang habt
4. Die vierte Zeile enthält 'x y': Die Startposition der Spielfigur
5. Dann folgen $n \cdot n$ Zeilen, die jeweils 'card_type object_id' enthalten (das Trennzeichen ist ein Leerzeichen)
 - card_type ist die ID der Karte
 - object_id identifiziert den gesuchten Informatiker. Ist der Wert $object_id \geq 0$, heißt das, daß der Informatiker als $object_id$ -tes gefunden werden soll. Ist der Wert -1 , so gibt es auf dieser Karte keinen Informatiker.

Ausgabe der Lösung

- Die Lösung soll so viele Zeilen haben, die ihr Spielzüge benötigt.
- Jeder Spielzug besteht aus `'id r|c num x y'`
 1. `id` ist die Kartenart, die eingeschoben werden soll. Sie muss „kompatibel“ mit dem tatsächlichen freien Kartentyp sein.
 2. `r` bedeutet Reihenverschieben, `c` Spaltenverschiebung
 3. `num` ist die Nummer der Spalte/Reihe (muss gerade sein, beginnt bei 1). Wenn `num` positiv ist, wird nach rechts/unten geschoben, sonst nach links/oben.
 4. `x y` Die Endposition der Spielfigur.
 5. Wir überprüfen, ob alle Spielzüge gültig sind und sie alle Informatiker in der richtigen Reihenfolge finden.

Note

`num` ist 1-indiziert!

Versionskontrolle

- Wir haben euch allen mercurial repositories aufgesetzt.
- Zu finden unter
`http://hackolaus.fachschaft.tf/hg/$GRUPPENNAME`
- Ein kleines Beispiel der Eingabe mit einer gültigen Lösung ist schon eingechekkt.
- Ihr benötigt nicht viele mercurial Kenntnisse...
 - `hg clone <url>` zum Auschecken
 - `hg pull` dann `hg update` zum Updaten
 - `hg commit -m <message>` und `hg push` zum Committen.
 - Wenn ihr branched oder es sonst wie kaputt bekommt... selbst Schuld und googeln ;)
- Wer noch nicht angemeldet ist, möge bitte gleich nach vorne kommen.