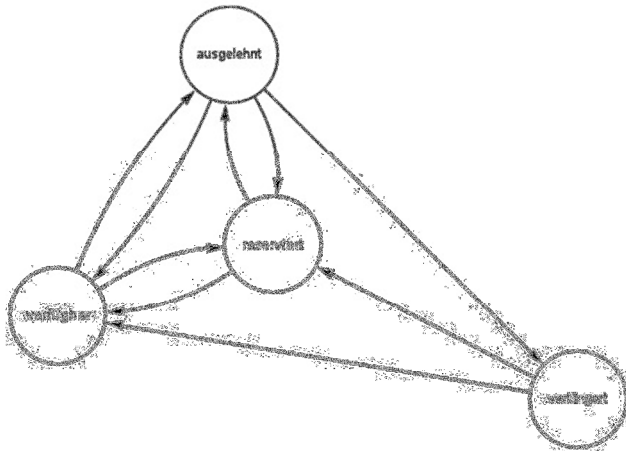


Automation: Automaten

Eine mögliche Lösung:



Aufgabe: Kara Programm lesen

Bei dieser eher einfachen Aufgabe spielt es keine Rolle, ob die Eckfelder wie unten doppelt nummeriert werden (zum Beispiel „6, 7“, „16, 17“) oder nicht):

	<div style="border: 1px solid black; padding: 5px;"> <p>suche</p> <table border="1"> <thead> <tr> <th colspan="3">Kara macht:</th> <th>Nächster Zustand:</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>no</td> <td>yes</td> <td>↑</td> <td>suche</td> </tr> <tr> <td>X</td> <td>yes</td> <td>no</td> <td>↻ ↑</td> <td>suche</td> </tr> <tr> <td>X</td> <td>yes</td> <td>yes</td> <td>↻</td> <td>suche</td> </tr> <tr> <td>X</td> <td>yes</td> <td>yes</td> <td>ⓘ</td> <td>Stop</td> </tr> </tbody> </table> </div>	Kara macht:			Nächster Zustand:	X	no	yes	↑	suche	X	yes	no	↻ ↑	suche	X	yes	yes	↻	suche	X	yes	yes	ⓘ	Stop
Kara macht:			Nächster Zustand:																						
X	no	yes	↑	suche																					
X	yes	no	↻ ↑	suche																					
X	yes	yes	↻	suche																					
X	yes	yes	ⓘ	Stop																					

Beschreiben Sie in zwei, drei Sätzen präzise, wie das Programm das Kleeblatt findet: Was ist die Idee hinter dem Programm? Ist das Programm effizient?

Das Programm lässt Kara den Bäumen rechts entlang laufen: Das Programm stellt sicher, dass immer auf Karas rechter Seite ein Baum ist.

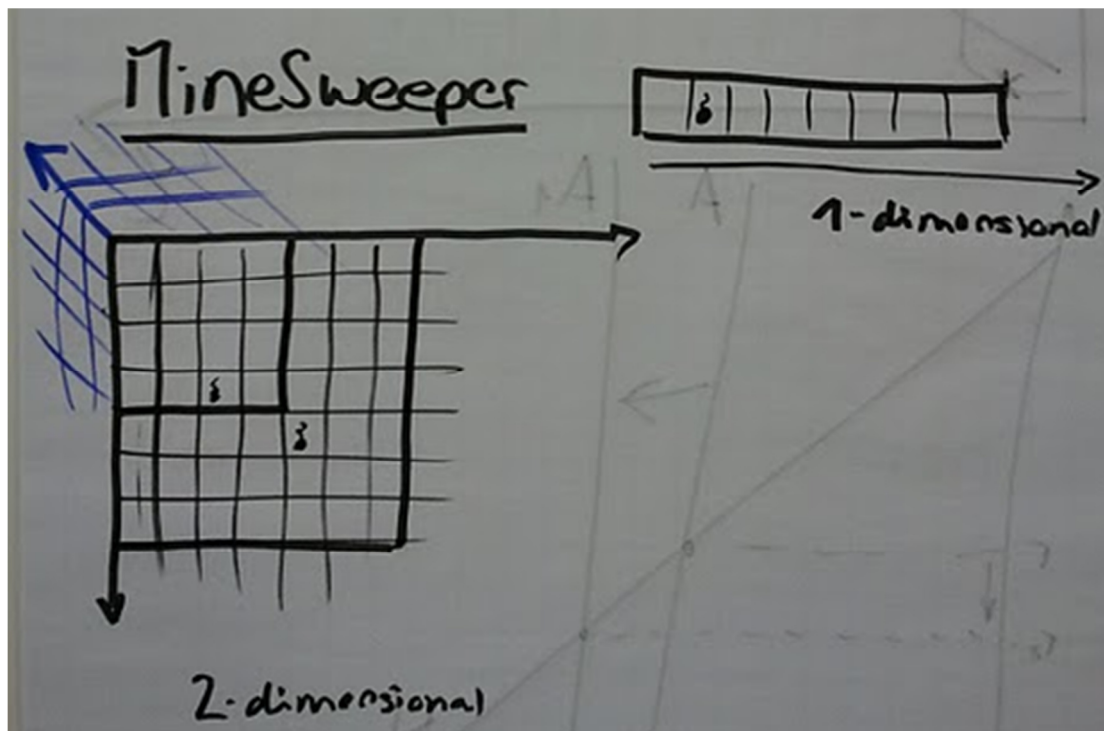
Daher ist das Programm nicht effizient: Es nutzt nur die Lücken zu seiner rechten Seite; an den Lücken zu seiner linken Seiten läuft er einfach vorbei.

Übrigens, schwieriger wäre die Frage: Würde ein Zustand für eine effiziente Lösung (nicht an Baumücken vorbeilaufen) ausreichen oder braucht es mehrere Zustände?

Programmierung: Arrays für Nicht-Informatiker erklärt

Eine mögliche Lösung; relevant ist bei denjenigen Arrays, die wir betrachtet haben, vor allem, dass der Inhalt über einen Index angesprochen wird. Der Fachbegriff Index ist dabei zweitrangig; der Index kann auch als Koordinate oder Adresse betrachtet werden.

Man kann sich Arrays vorstellen wie Tabellen in Excel. Wir könnten zum Beispiel ein Excel erstellen für das Spiel Minesweeper. Das sähe dann in etwa so aus:



In so einer Tabelle könnte gespeichert werden, an welchen Stellen Bomben liegen. Diese Tabelle wäre in einem Programm ein Array. In Excel würde man die einzelnen Zellen ansprechen als A1, A2, ..., B1, B2, ... etc. In einem Programm würde man dazu ganze Zahlen verwenden: 0,0; 0,1; ...; 1,0; 1,1; ... Der Zugriff auf ein Feld erfolgt dann über diese Zahlen (Index).

Im Unterschied zu Excels Tabellen können Arrays auch einfach eindimensional sein (siehe Skizze oben rechts). Das wäre dann einfach eine einzelne Zeile (oder eine einzelne Spalte) in Excel. Oder Arrays können sogar drei- oder mehrdimensional sein – das sprengt den Vergleich mit Excel ☺