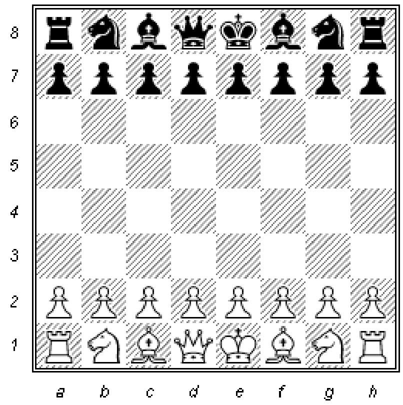
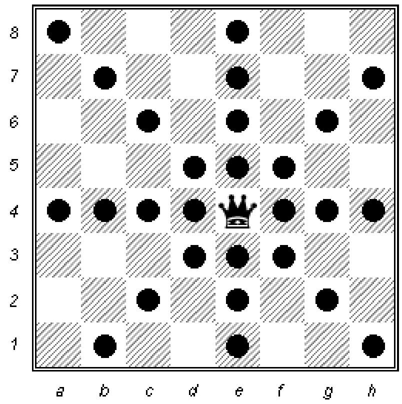
Beispiel: Das Acht-Damen-Problem



Das Damenproblem ist ein klassisches mathematisches Problem, das der bayrische Schachmeister Max Bezzel 1848 in einer Schachzeitschrift veröffentlichte.

Die Aufgabe besteht darin, auf einem Schachbrett (ein Quadrat mit acht Zeilen und acht Spalten) acht Damen so zu platzieren, dass keine Dame eine andere „bedroht".
Es sollen nicht nur eine, sondern alle Möglichkeiten gefunden werden, die acht Damen auf diese Weise zu platzieren.



Erläuterung:
Eine Dame ist eine Schachfigur, die wie folgt ziehen kann:
Auf jedes Feld in allen vier Himmelsrichtungen, sowie auf jedes Feld diagonal von ihrer Position aus (siehe Abbildung).
Wenn eine Dame auf einem ansonsten leeren Schachbrett steht, „bedroht" sie alle Felder, die mit einem Punkt markiert sind.
Auf diese Felder dürfte man daher keine weitere Dame stellen.

1874 bewies der Mathematiker James Glaisher, dass es bei einem 8x8-Schachbrett 92 verschiedene Möglichkeiten gibt, die acht Damen in der geforderten Weise zu platzieren.

Aufgabe 1

a) Löse das Problem **zeichnerisch**. Beginne zunächst mit einem kleineren Schachbrett mit 4 Zeilen und 4 Spalten und platziere vier Damen so, dass keine Dame eine der anderen bedroht. Versuche es dann mit einem 5x5-Schachbrett und 5 Damen, und dann für 6x6 mit 6 Damen.

b) Formuliere einen **Algorithmus**, wie das Problem <u>systematisch</u> gelöst werden kann.
Finde also eine Vorgehensweise, die sich mit einem Computer umsetzen lässt.
Es macht nichts, wenn die Lösung viel Zeit brauchen würde.

Der Algorithmus soll als Idee formuliert werden, es geht nicht um die genaue Implementierung!

Lösung: Algorithmus für das Acht-Damen-Problem

Um das Problem systematisch zu lösen, überlegen wir zunächst folgende hilfreiche **Annahme**:

Eine Dame blockiert immer eine ganze Zeile, es können also nie zwei Damen in der gleichen Zeile stehen. Da es acht Zeilen gibt, und acht Damen platziert werden müssen, muss auf jeder Zeile genau eine Dame stehen.

Algorithmus:

1. Platziere eine Dame in der ersten Zeile auf dem erstmöglichen Feld, also ganz links.
   Platziere nacheinander in jeder weiteren Zeile eine Dame, immer auf dem erstmöglichen Feld, also so weit links wie möglich, so dass sie nicht von bereits platzierten Damen bedroht wird.

2. Wenn man auf der letzten Reihe eine Dame platzieren kann, ist eine Lösung gefunden.
   (Dann muss aber noch nach weiteren Möglichkeiten gesucht werden.)

3. Wenn es in einer Zeile keine Möglichkeit gibt, weil bereits alle Felder bedroht sind, gehe eine Zeile zurück. Schiebe die Dame auf dieser Zeile weiter nach rechts auf das nächste mögliche Feld. Dann probiere es weiter in den nächsten Zeilen.

4. Falls es auch in der vorigen Zeile keine Möglichkeit mehr gibt, gehe noch weiter zurück bis zu einer Zeile, wo es noch Möglichkeiten gibt, die nicht probiert wurden.

5. Der Algorithmus endet, wenn auch in der ersten Zeile keine Möglichkeiten mehr ausprobiert werden können.

Anwendung eines Stacks für den Algorithmus

In Backtracking-Algorithmen nutzt man einen Stack, um die bereits ausprobierten Möglichkeiten festzuhalten. Jeder neue Schritt wird auf den Stack gelegt. Beim Zurückgehen nimmt man den Schritt wieder vom Stack herunter.

Im Beispiel des Schachbretts mit 4x4 Feldern würde man einen Stack wie folgt verwenden:

(leer)	Das Schachbrett ist am Anfang leer
A1	Das erste mögliche Feld auf der 1. Reihe ist A1
A1 – C2	Das erste mögliche Feld auf der 2. Reihe ist C2
A1	In der 3. Zeile gibt es keine freien Felder mehr. Also zurück zur 2.
A1 – D2	Probiere das nächste freie Feld in der 2. Zeile.
A1 – D2 – B3	Das einzige freie Feld auf der 3. Reihe ist B3.
(leer)	Da es auf der 4. Reihe keine freien Felder gibt, gehe zurück. In der 2. Zeile wurden bereits alle Felder probiert, also zurück zur 1.
B1	Probiere das nächste freie Feld in der 1. Zeile.
B1 – D2	Das einzige freie Feld auf der 2. Zeile ist D2.
B1 – D2 – A3	Das einzige freie Feld auf der 3. Zeile ist A3.
B1 – D2 – A3 – C4	Auch in der 4. Zeile gibt es ein freies Feld: C4. Damit ist eine der möglichen Lösungen gefunden.

Backtracking allgemein

Backtracking-Algorithmen bestehen aus einer Kette von Entscheidungen, bei denen man jedesmal die Wahl zwischen zwei oder mehr Möglichkeiten hat. Je nach dem, wie man sich an jeder Stelle entscheidet, findet man eine Lösung des Problems, oder kommt an einer Stelle nicht weiter und muss zurück gehen.

Allgemein formuliert man Backtracking-Algorithmen wie folgt:

1. Lege eine **Reihenfolge** fest, in der bei jeder Entscheidung die Möglichkeiten probiert werden.

2. Bei der ersten **Entscheidung** wähle die erste Möglichkeit entsprechend der Reihenfolge.
   Auch bei der nächsten Entscheidung wieder die erste Möglichkeit, und so weiter.

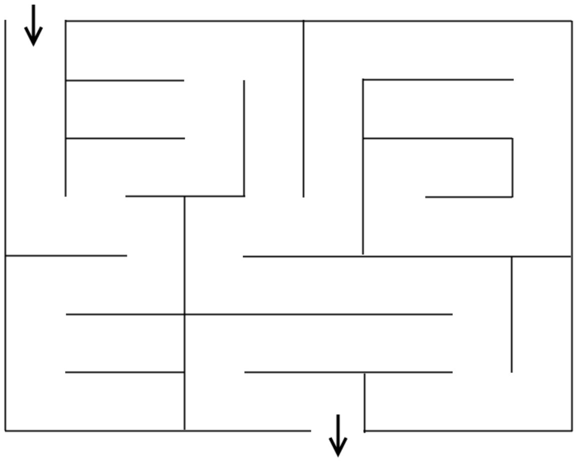
3. Wenn es an einer Stelle keine Möglichkeit mehr gibt, **gehe zurück** bis zu einer Entscheidung, wo noch nicht alle Möglichkeit ausprobiert wurden.
   Wähle dort die nächste Möglichkeit, und gehe wieder weiter vor.

4. **Wiederhole** diesen Prozess, bis du eine (oder alle) Lösungen gefunden hast,
   oder es keine Möglichkeiten zum Ausprobieren mehr gibt.

Der Algorithmus **endet**, wenn eine Lösung, oder wenn alle Lösungen gefunden wurden, oder alle Möglichkeiten ausprobiert wurden, ohne dass eine Lösung gefunden wurde.
Falls es mehrere Lösungen gibt, kann man ggf. die beste Möglichkeit auswählen, z.B. den kürzesten Weg zwischen zwei Orten.

Verständnisfrage: Woran erkennt man, dass alle Möglichkeiten gefunden wurden?

Aufgabe 2

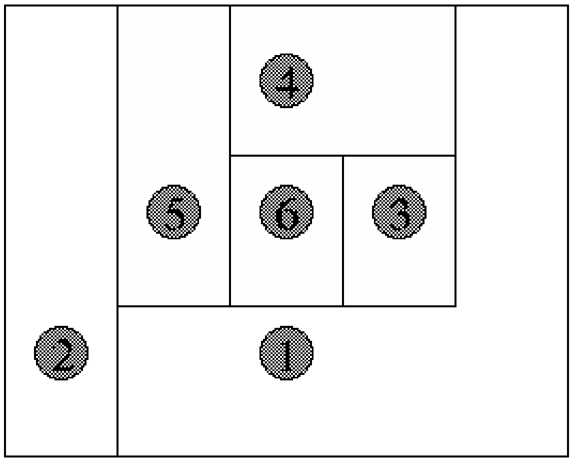


Gegeben sei ein beliebiges Labyrinth mit einem Eingang und einem Ausgang.
Es hat keine „zyklischen“ Wege, mit denen man im Kreis gehen könnte.

a) Formuliere einen Backtracking-Algorithmus, mit dem man den Ausgang kann (nicht den „Rechte-Hand-Algorithmus“). Orientiere dich an den Formulierungen für das Acht-Damen-Problem.

b) Führe den Algorithmus mithilfe eines Stacks für das hier abgebildetet Labyrinth durch. Notiere jedes Vor- und Zurückgehen, analog zur Lösung des Damen-Problems auf Seite 2.

Aufgabe 3



Das „Vier-Farben-Problem“ besteht darin, auf einer beliebigen Landkarte alle Länder mit je einer Farbe einzufärben. Die Schwierigkeit besteht darin, dass nur vier verschiedene Farben zur Verfügung stehen, und Nachbarländer nie die gleiche Farbe haben sollen.

Hinweis: Länder, die sich nur an einem Punkt berühren, gelten hier nicht als benachbart.

a) Formuliere einen Backtracking-Algorithmus, um eine beliebige Landkarte schrittweise mit den genannten Einschränkungen einzufärben.

b) Führe deinen Algorithmus mithilfe eines Stacks für die hier abgebildete „Landkarte“ durch. Notiere jedes Vor- und Zurückgehen, analog zur Lösung des Damen-Problems.

Zur Geschichte des Vier-Farben-Problems:

Das Problem wurde 1840 zum ersten Mal in einer Mathematik-Vorlesung erwähnt. Der erste Beweis, dass es möglich sei, jede beliebige Karte mit vier Farben einzufärben, erwies sich als fehlerhaft. Erst in den 1960er und 70er-Jahren entwickelte der Mathematiker Heinrich Heesch ein computergestütztes Verfahren, das eine ganze Reihe möglicher Spezialfälle durch Ausprobieren abtesten sollte. Ihm stand damals jedoch nicht die nötige Rechenleistung zur Verfügung, sein Verfahren durchzuführen. 1976 gelang es dann zwei Mathematikern der Universität Illinois, mithilfe eines Computerprogramms zu beweisen, dass es für jede beliebige Karte eine Einfärbung mit vier Farben gibt. Das Programm brauchte damals über 1000 Stunden Zeit.