Tutorium Programmiersprachen 2

30. Oktober 2006

Tutor: Arne Johannessen

Fragen bitte sofort stellen

falls ich zu schnell spreche, bitte unterbrechen ☺

Teil 0: Organisation

- Überzeugungen
- Quellcode für die Übungsaufgaben
  - selber ’rauskopieren macht häufig Probleme
    - ↳ Quellcode ist online als .java-Datei
- beim Verändern von Code *immer* eigenen Namen und Datum hinzufügen!

http://www.home.hs-karlsruhe.de/~joar0011/tutorium

Tutorium bei Arne Johannessen - Programmiersprachen 2

Übersicht
Termine
Bewertung
Quellcode für die Aufgaben
Kontakt
Datenschutz Impressum

Quellcode für die Übungsaufgaben

Die Übungsaufgaben bauen in diesem Semester auf einer Reihe von Quellcode-Vorlagen auf, die in den Vorlesungsfolien (aus Herrn Bürgs Downloadbereich) enthalten sind. Da das Herauskopieren oder Abtippen des Quellcodes nicht immer zuverlässig ist, stelle ich hier diesen Quellcode zum direkten Download zur Verfügung.

1. `Telefon.java`
2. `LAF.java`

(Rest kommt später)

Der Quellcode entspricht bis auf minimalste Änderungen genau dem Original aus den Folien, kann also bedenkenlos verwendet werden.

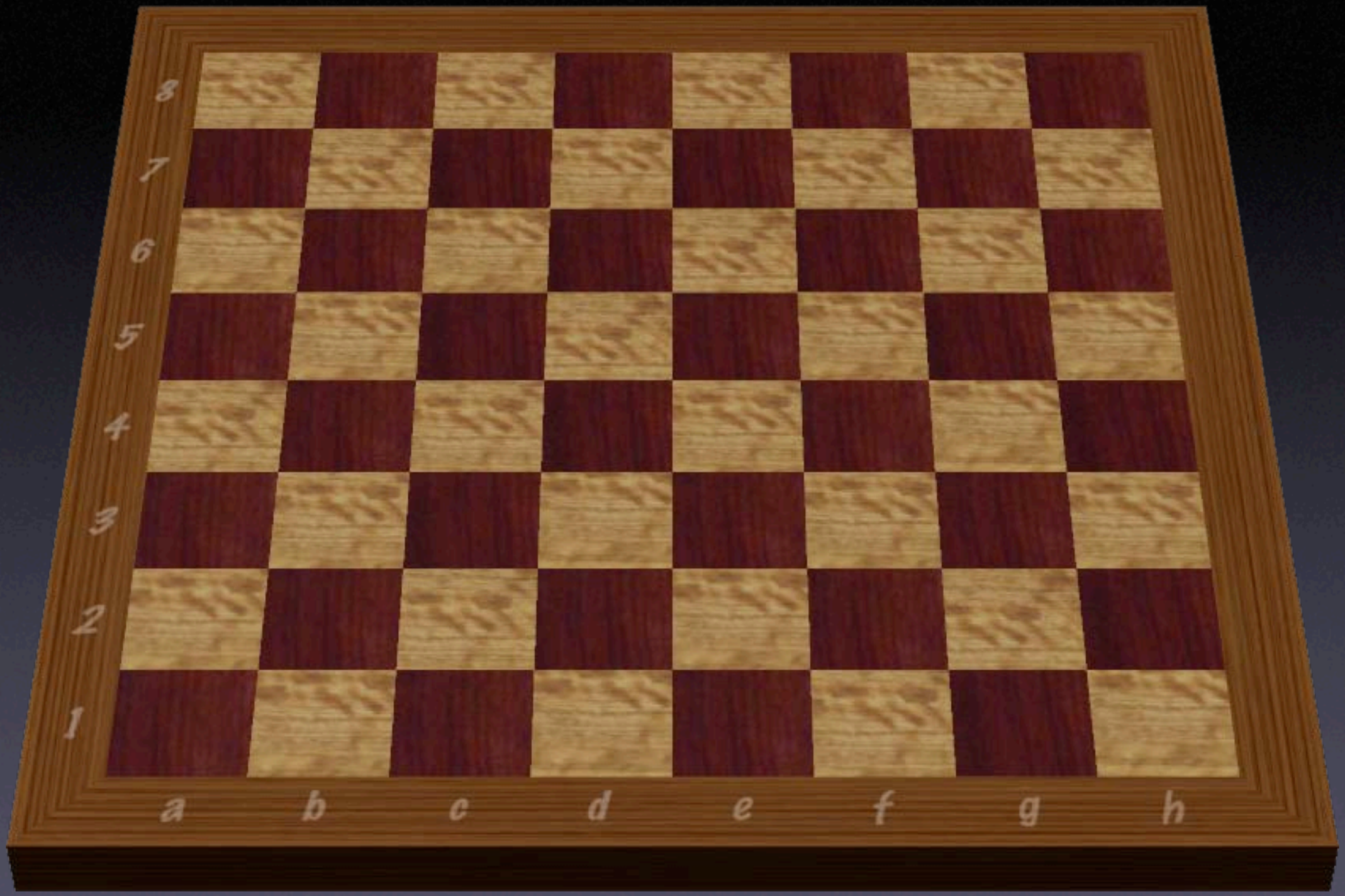
Arne Johannessen, 30. Oktober 2006

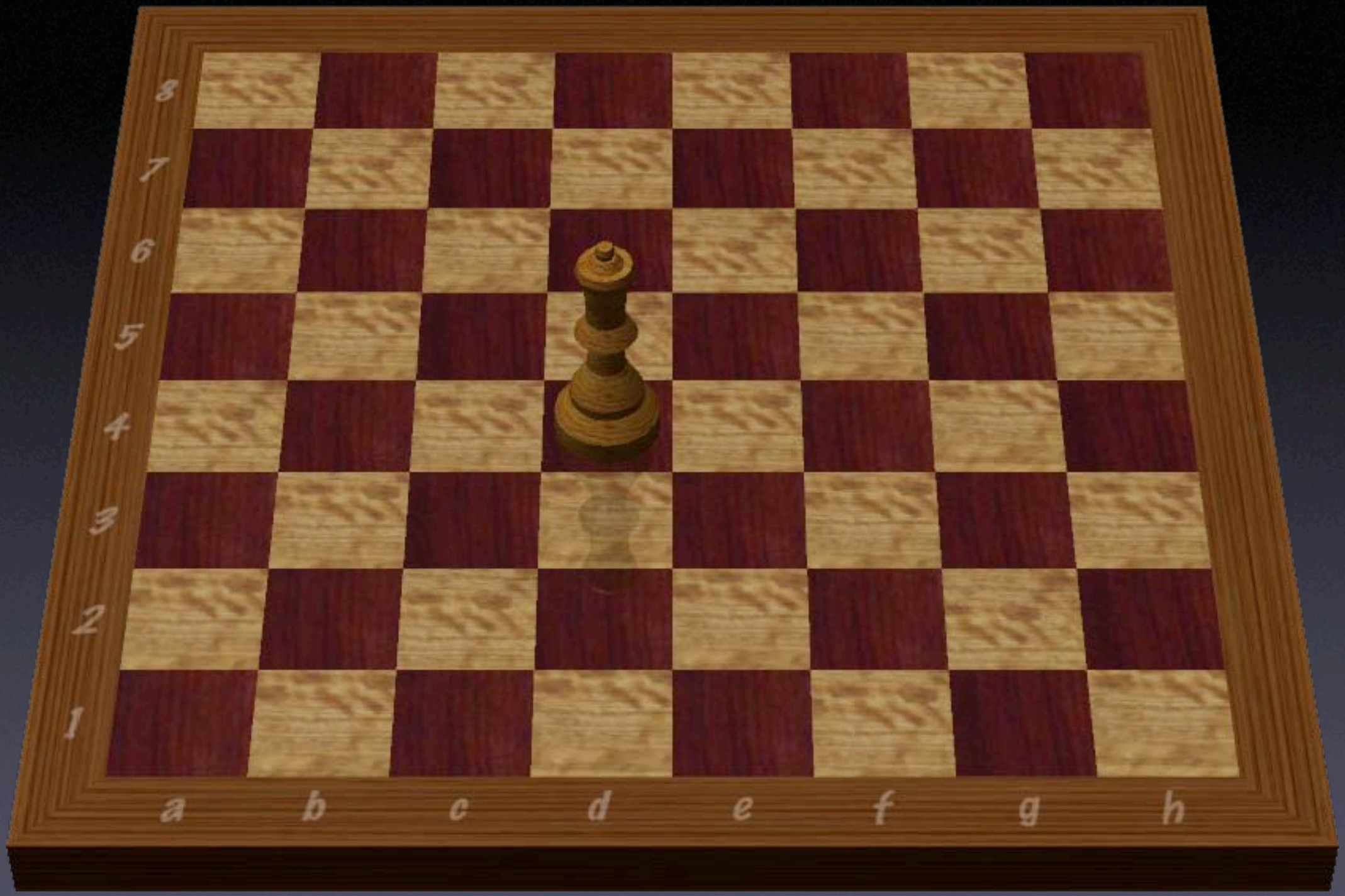
Teil 1: ein Schachproblem

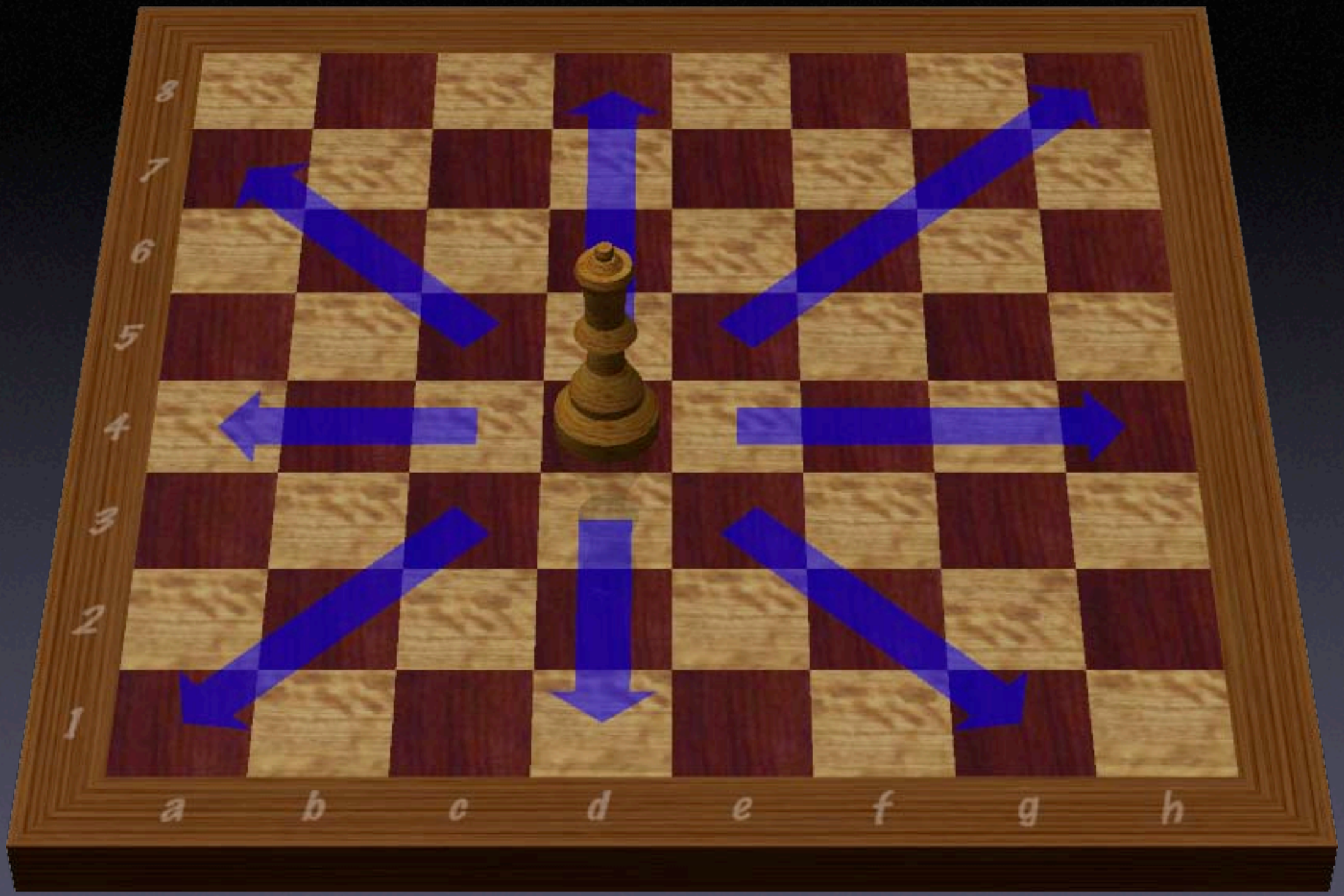
- dies ist ein Schachproblem, für das eine Berechnung der Lösung durch den Computer besonders einfach als Programm umzusetzen ist

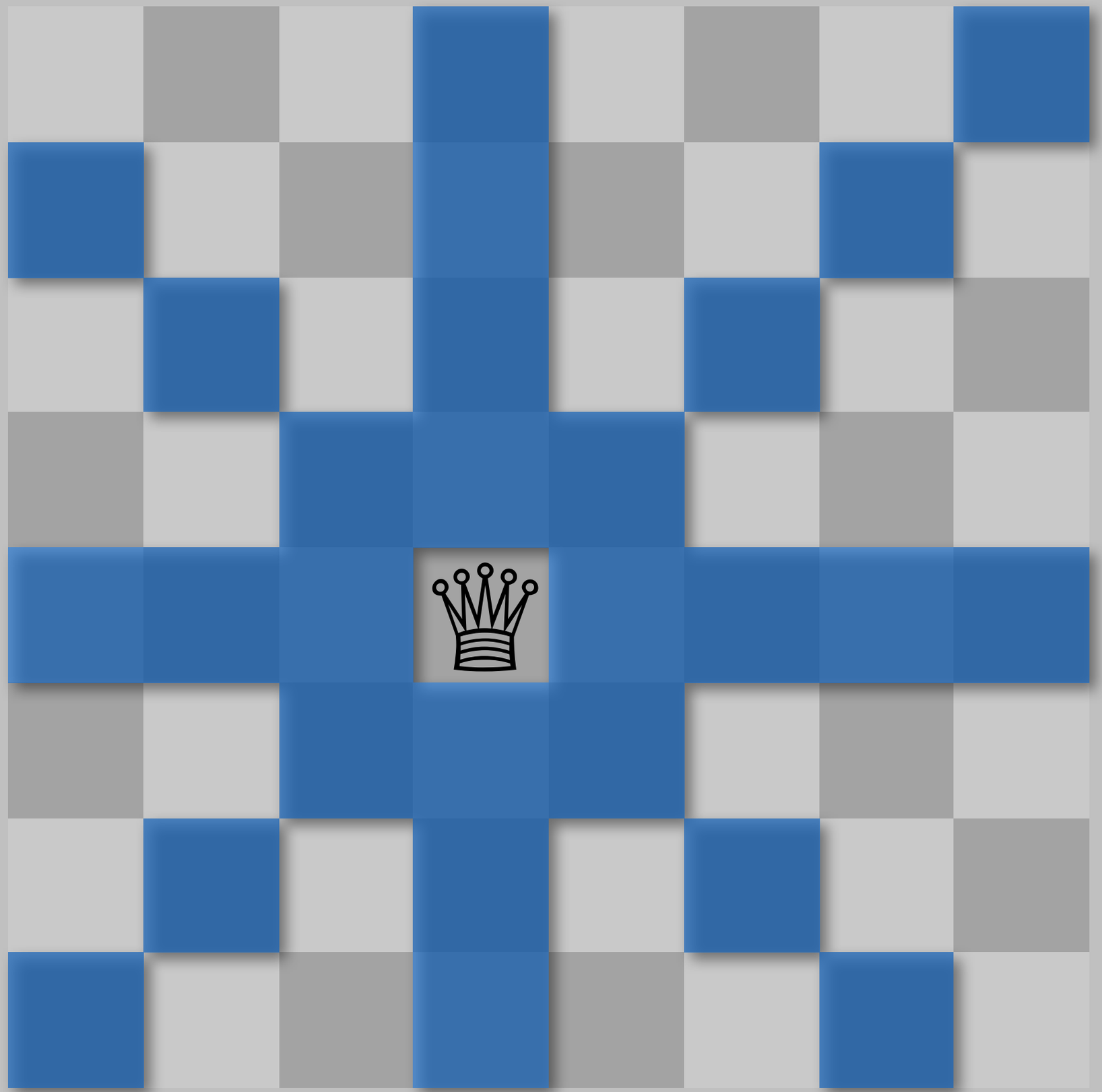
„äh – ‚Schach..?'"


8
7
6
5
4
3
2
1
a
b
c
d
e
f
g
h


8
7
6
5
4
3
2
1
a
b
c
d
e
f
g
h


8
7
6
5
4
3
2
1
a
b
c
d
e
f
g
h


8
7
6
5
4
3
2
1
a
b
c
d
e
f
g
h



Schachaufgabe

„Acht-Damen-Problem“

Es sind acht Damen so auf dem Schachbrett zu platzieren, dass keine die andere deckt.

Acht-Damen-Problem:

Es sind acht Damen so auf dem Schachbrett zu platzieren, dass keine die andere deckt.

64 Felder

- acht Spalten („Linien“)
- acht Zeilen („Reihen“)

$\Rightarrow$ genau eine Dame je Spalte und Zeile!

Acht-Damen-Problem:

Es sind acht Damen so auf dem Schachbrett zu platzieren, dass keine die andere deckt.

Acht-Damen-Problem:

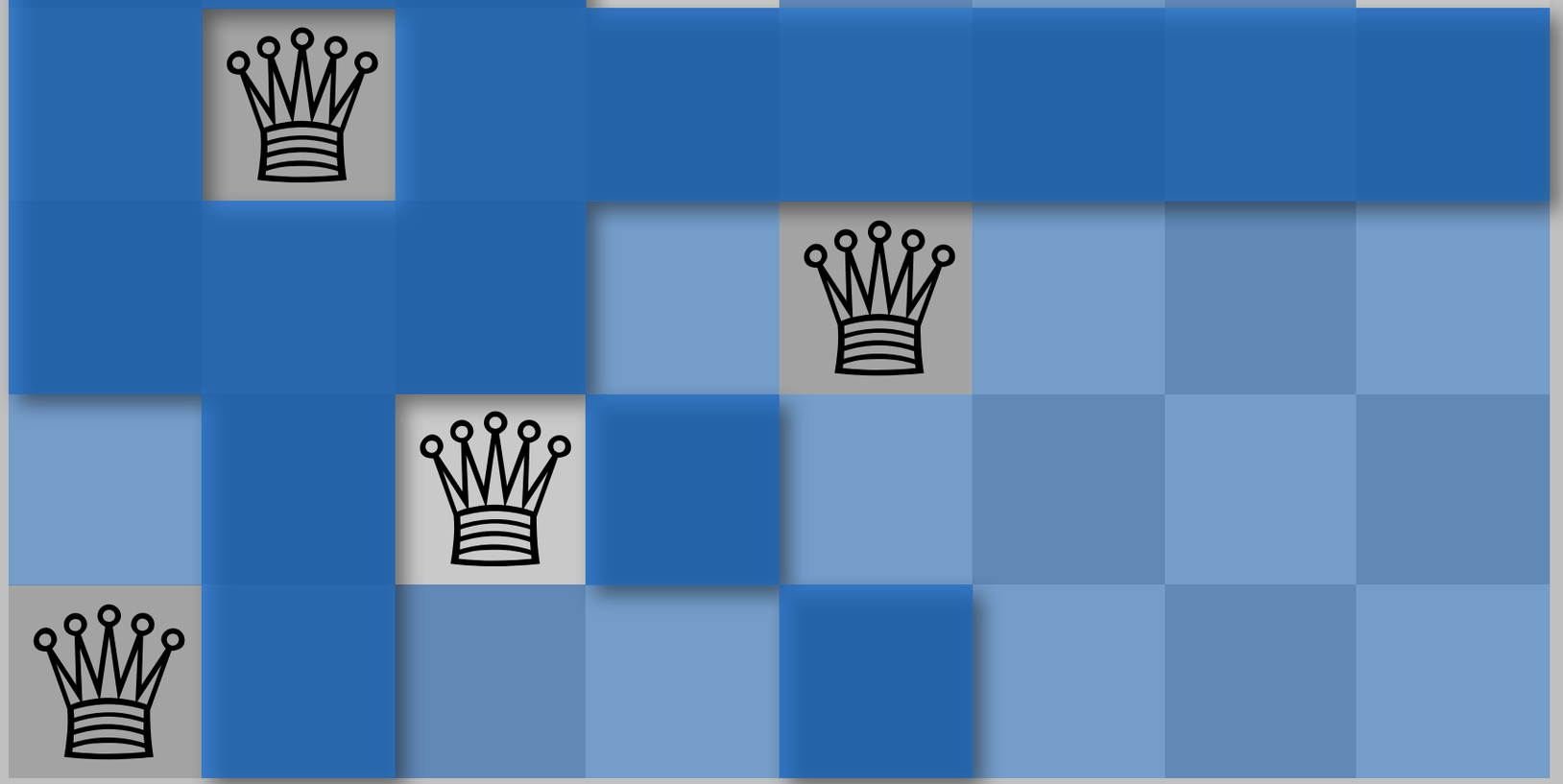
Es sind acht Damen so auf dem Schachbrett zu platzieren, dass keine die andere deckt.

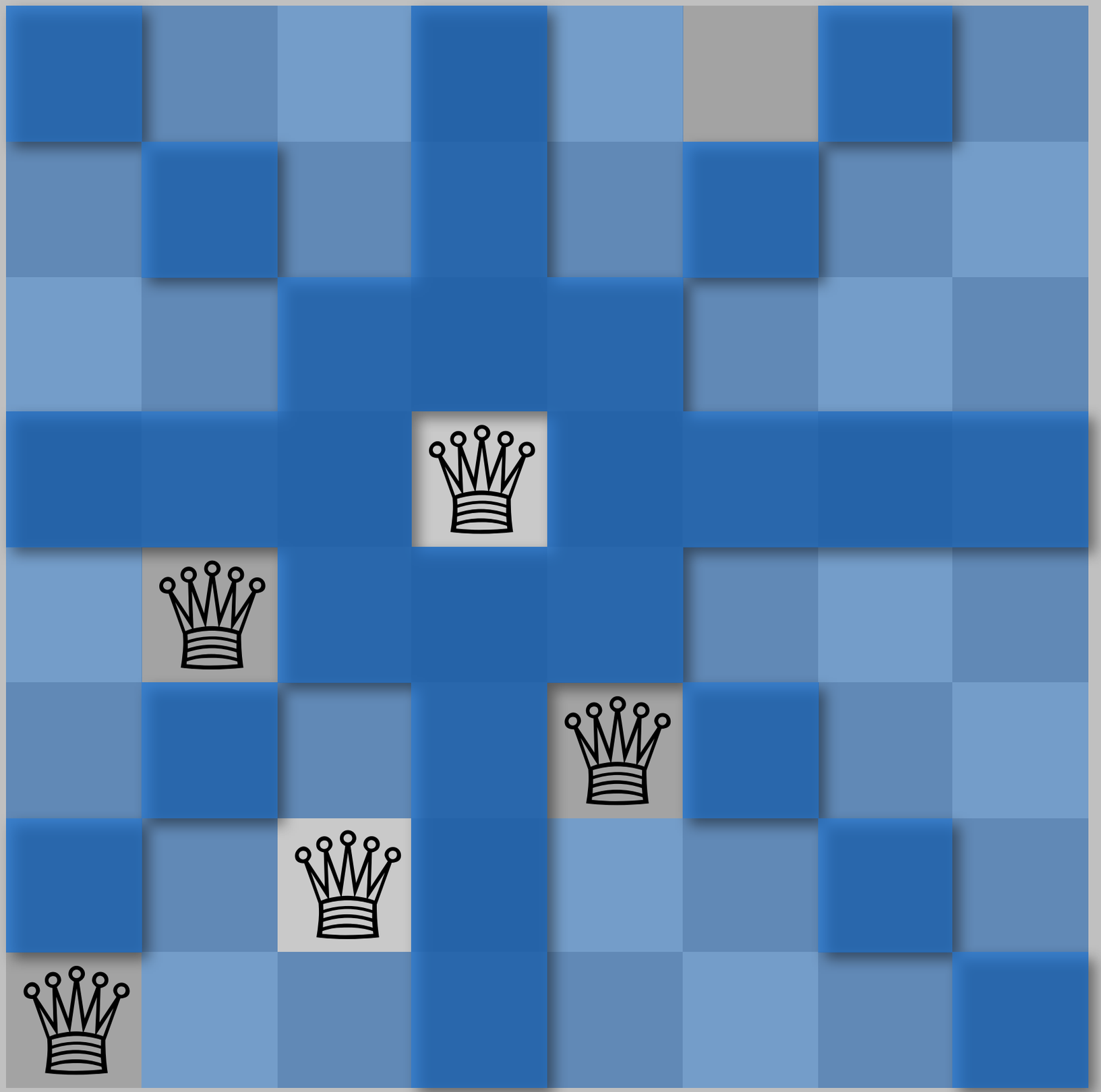
Acht-Damen-Problem:

Es sind acht Damen so auf dem Schachbrett zu platzieren, dass keine die andere deckt.

Acht-Damen-Problem:

Es sind acht Damen so auf dem Schachbrett zu platzieren, dass keine die andere deckt.

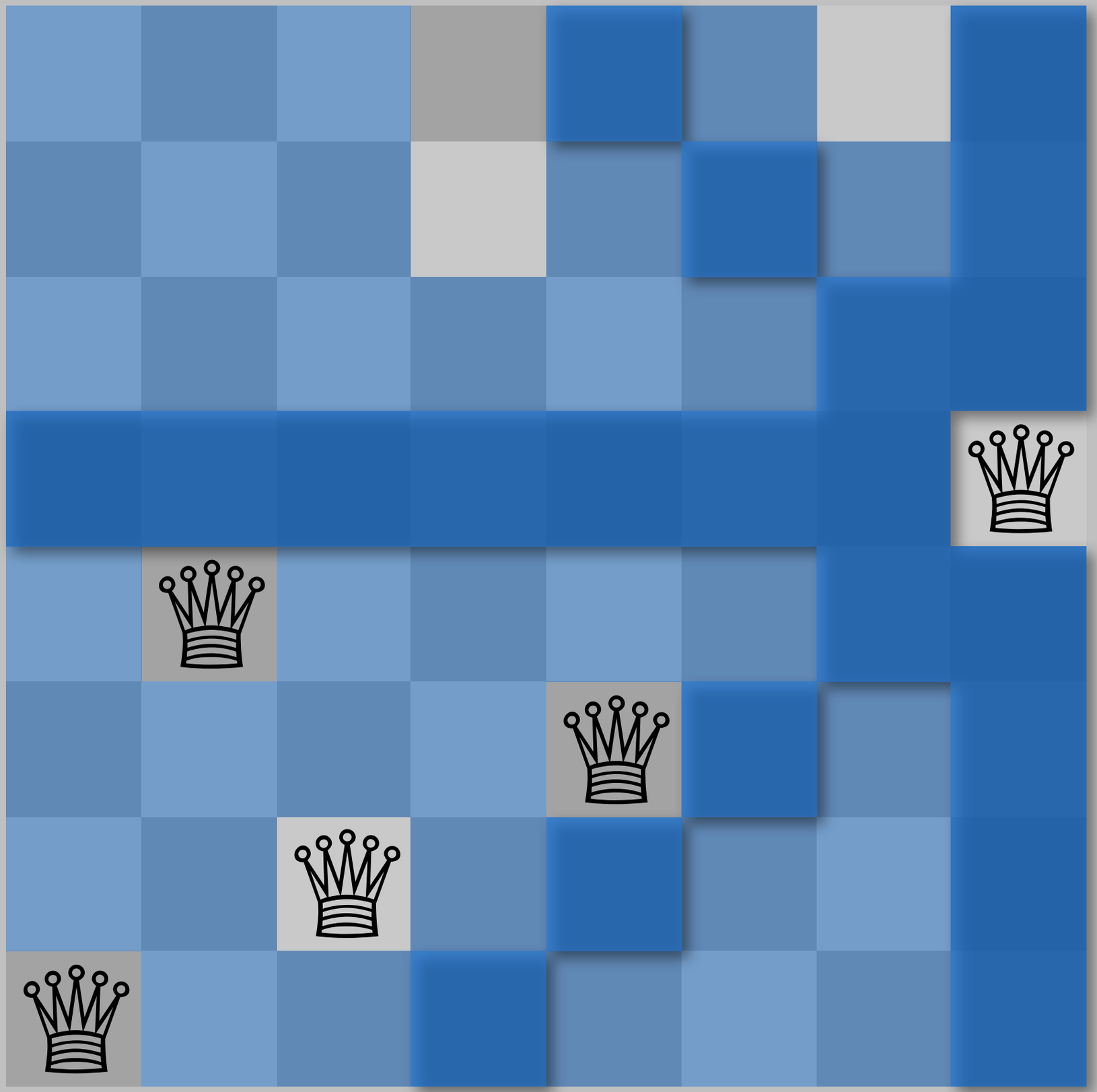




Problem:
alle Felder in Reihe 6 sind gedeckt!
$\Rightarrow$ kein Platz für weitere Dame

Lösungsweg:

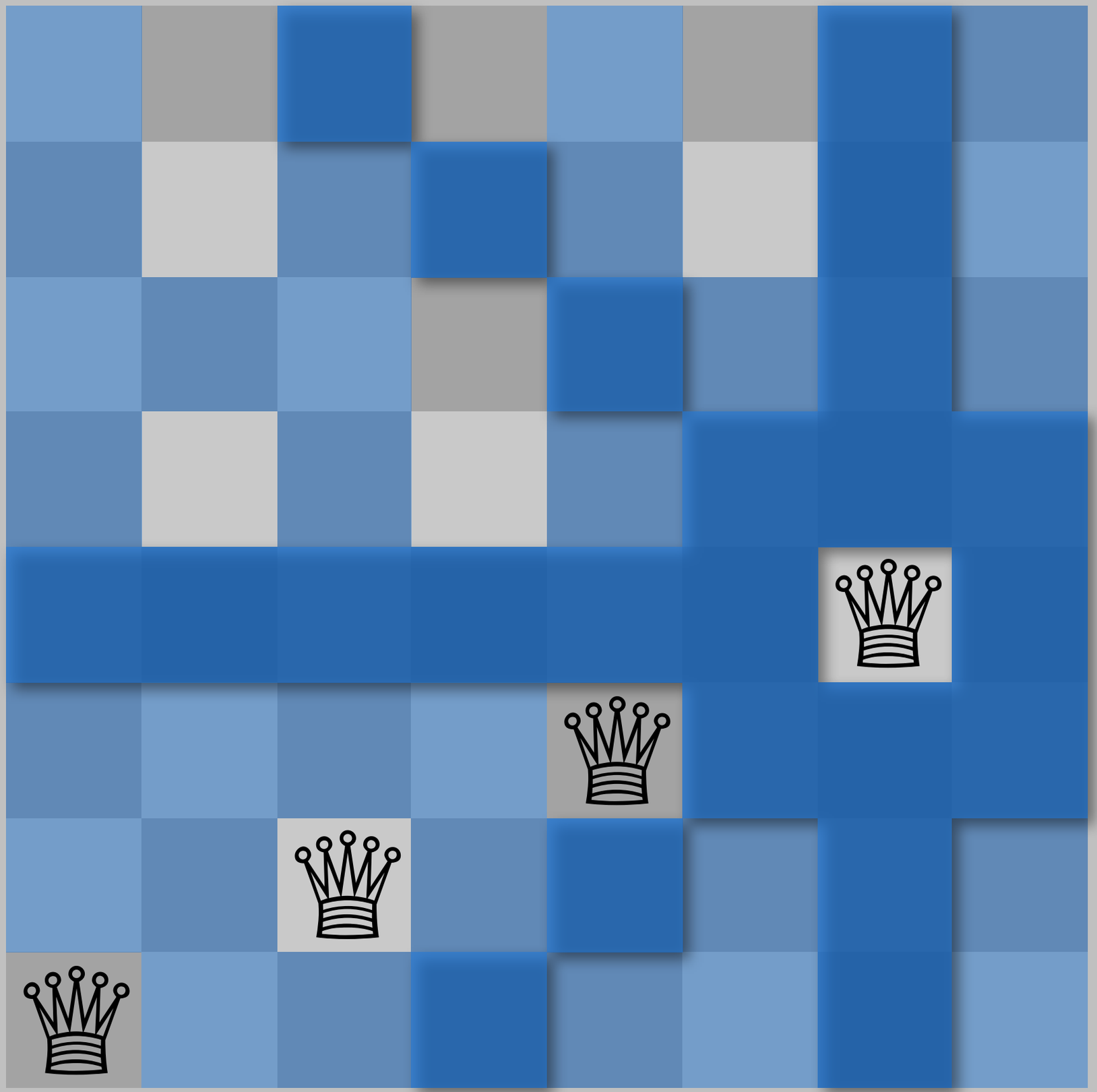
Backtracking (zurückziehen)



gleiches Problem in Reihe 6

einmal zurückziehen reicht nicht

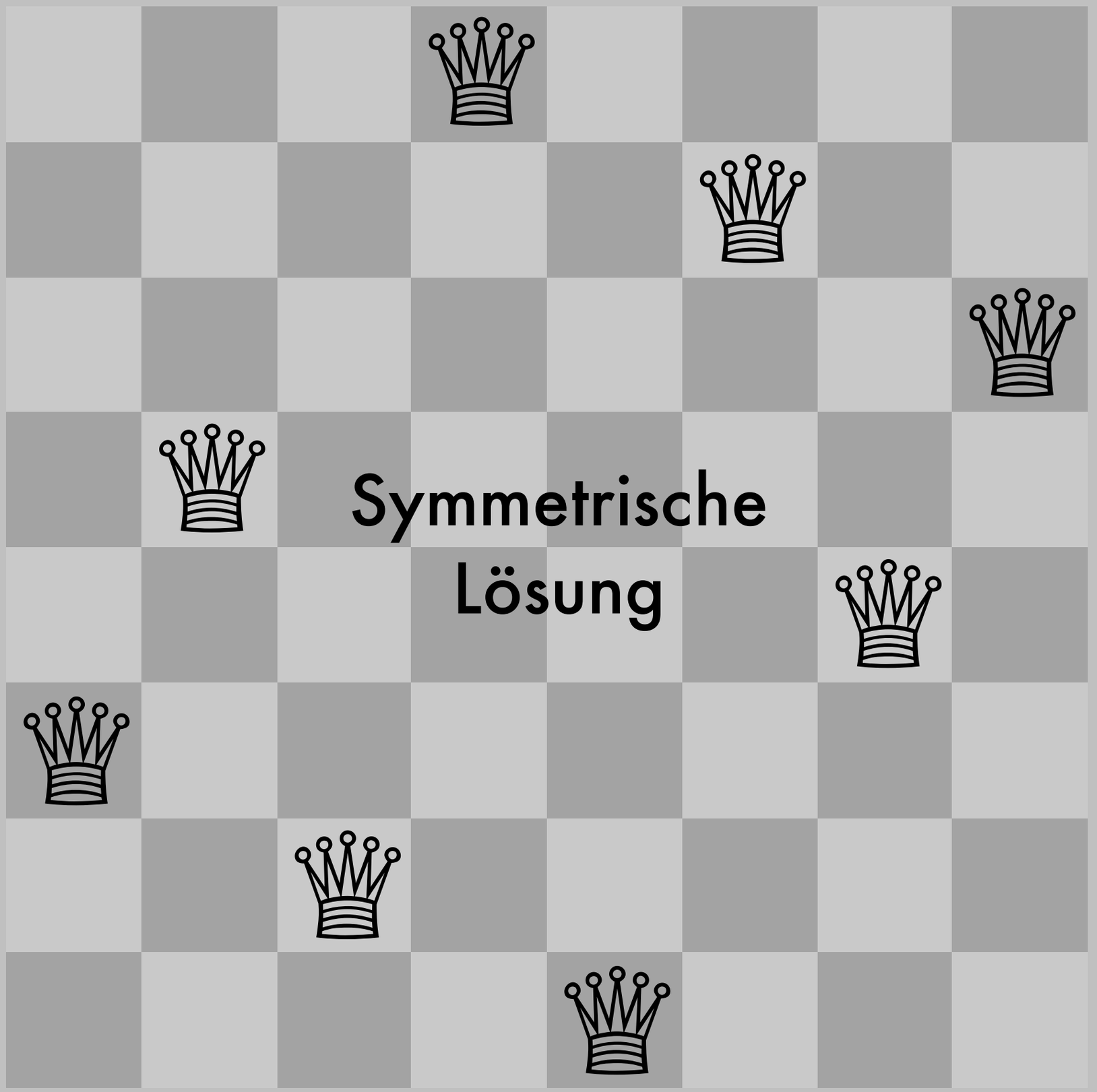
⇒ auch Reihe 4 zurückziehen!




... und so weiter!

Acht-Damen-Problem

- prima mit Backtracking zu lösen
- Lösungsverfahren probiert mit brutaler Gewalt alle existierenden Stellungen durch:
  - *Brute-Force*–Lösungsverfahren
- es existiert mehr als eine Lösung
  - Anzahl hängt von Zählweise ab
  - die meisten Stellungen lassen sich durch Drehen des Brettes aus anderen Stellungen erzeugen


Symmetrische
Lösung

Teil 2: Schachproblem-Lösung mit Java

- Aufgabe eines Informatikers
  - einschließlich Geoinformatiker = Geomatiker!
- eigenständiges Entwickeln von zur Problemlösung geeigneten *Algorithmen und Datenstrukturen* nötig
  - vgl. Vorlesung „Algorithmen und Datenstrukturen 2“
- selbstständige Lösung einer Aufgabe dieser Art ist *nicht* Prüfungsstoff in „Informatik 2“!
  - daher:
    - Aufgabe freiwillig
    - Anleitung durch mich

Teil 2: Schachproblem-Lösung mit Java (2)

- Lernziele:
  - Auffrischen der Grundlagen aus „Programmiersprachen 1“ letztes Semester
  - insbesondere Festigen von:
    - Begriffen
    - Datentypen
    - Werte von Ausdrücken (Operatoren)
    - *Top-Down–Design* (Methoden)
    - Objektorientiertheit (Klassen)
- *nicht* Lernziel:
  - grafische Ein- und Ausgabe

teile und herrsche *(Divide and Conquer)*

- Zerlegen eines Problems in kleinere (einfacher zu lösende) Teilprobleme
- Beispiele:
  - Quicksort (vgl. „Algorithmen und Datenstrukturen 1“)
  - Top-Down–Design
- bei objektorientierter Programmierung in Java:
  - Lösen der Teilprobleme mit eigenständigen Objekten, Klassen und Methoden

„Acht Damen“ teilen

- Teilprobleme:
  - Koordinaten auf dem Schachbrett ✓
  - Datenstruktur ✓
    - zwei-dimensionaler Array eines neu anzulegenden Datentyps für Schachfiguren
      - ↳ **z. B.** `Schachfigur[][] schachbrett;`
        mit `class Schachfigur { … }`
        und `class Dame extends Schachfigur { … }`
    - **(alternativ wäre hier ebenfalls möglich:** `boolean[][]`**)**
  - …

Fortsetzung folgt..!