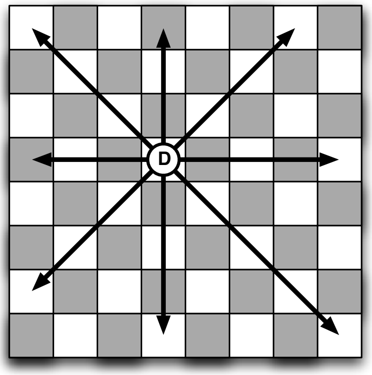
27. Bundeswettbewerb Informatik 2008/2009 Endrunde 22.-25.9.2009

Das n-Damen-Problem

Gruppenarbeit 1. Tag

Sei n eine beliebige natürliche Zahl ≥ 2. Beim *n-Damen-Problem* ist ein $n \times n$-Schachbrett gegeben. Das Ziel ist, auf diesem Schachbrett n Damen so zu positionieren, dass keine der Damen eine andere nach den Schachregeln schlagen kann (zur Erinnerung: beim Schach kann eine Dame beliebig weit in horizontaler Richtung, in vertikaler Richtung oder in diagonaler Richtung ziehen, siehe Abbildung unten).



Ziel dieser Aufgabe ist, einen Algorithmus zu entwerfen, der bei Eingabe einer beliebigen Zahl n alle Möglichkeiten auflistet, wie n Damen auf dem $n \times n$-Schachbrett so positioniert werden können, dass keine der Damen eine andere Dame schlagen kann.

Natürlich soll der Algorithmus zum einen möglichst zeiteffizient sein und zum anderen möglichst wenig Speicherplatz verbrauchen.

Als „Zwischenschritte" zum Entwickeln des Algorithmus lösen Sie bitte die folgenden Aufgaben:

1. Geben Sie für $n = 2$, 3, 4, und 5 jeweils alle möglichen Lösungen des n-Damen-Problems an.

2. Eine nahe liegende Art, das n-Damen-Problem zu lösen, ist wie folgt:

Eine *Aufstellung* ist ein Boolesches Array A der Größe $n \times n$, so dass der Eintrag $A[i, j]$ angibt, ob eine Dame im Feld (i, j) in der i-ten Zeile und j-ten Spalte des Schachbretts steht.

Mit einem *Backtracking-Algorithmus* kann man das n-Damen-Problem dann folgendermaßen lösen: Durchlaufe das Schachbrett spaltenweise von links nach rechts und versuche, in jeder Spalte genau eine Dame auf einem noch nicht bedrohten Feld zu platzieren. Falls eine Dame nicht platziert werden kann, nimm die zuletzt platzierte Dame weg und bewege sie weiter.

Arbeiten Sie die Details aus und geben Sie

(I) einen Algorithmus an, der bei Eingabe einer Zahl $n \geq 2$ eine Lösung des n-Damen-Problems ausgibt bzw. die Information „es gibt keine Lösung", falls es keine Lösung gibt, und

(II) einen Algorithmus an, der bei Eingabe einer Zahl $n \geq 2$ *alle* Lösungen des n-Damen-Problems ausgibt (bzw. die Information „es gibt keine Lösung", falls es keine Lösung gibt).

Gehen Sie dabei insbesondere darauf ein, wie jede der drei Operationen

(P) Platzieren einer Dame

(E) Entfernen einer Dame

(Ü) Überprüfen, ob ein Feld bedroht ist

realisiert wird. Geben Sie auch an, welchen Zeitaufwand jede der drei Operationen erfordert und wie oft jede der drei Operationen von Ihrem Backtracking-Algorithmus aufgerufen wird.

3. Ein Nachteil der in (2) erarbeiteten Algorithmen ist, dass sie Speicherplatz in Höhe von $O(n^2)$ vielen Bits benötigen.

Entwickeln Sie Algorithmen für (I) und (II), die an Stelle des Booleschen Arrays A der Größe $n \times n$ ein 1-dimensionales Array S der Größe n benutzt, dessen Einträge Zahlen aus der Menge $\{0, 1, \ldots, n\}$ sind. Der Eintrag $S[j] = i$ für $i \geq 1$ soll dann besagen, dass sich in Spalte j eine Dame in Zeile i befindet. Der Eintrag $S[j] = 0$ soll besagen, dass sich in Spalte j (noch) keine Dame befindet.

Gehen Sie auch hier insbesondere auf den durch die drei Operationen (P), (E) und (Ü) verursachten Zeitaufwand ein.

4. Ein weiterer Nachteil der Algorithmen aus (2) und (3) ist, dass diejenige der drei Operationen (P), (E) und (Ü), die am häufigsten aufgerufen wird, einen recht großen Zeitaufwand erfordert.

Entwickeln Sie eine Datenstruktur, deren Speicherplatz nur um einen konstanten Faktor größer ist als bei (3), bei der aber jede der drei Operationen (P), (E) und (Ü) in konstanter Zeit (also in Zeit $O(1)$) durchgeführt werden kann. Nutzen Sie diese Datenstruktur, um Algorithmen für (I) und (II) zu entwickeln.

5. Versuchen Sie, Ihre Algorithmen zum Lösen des n-Damen-Problems weiter zu verbessern: z.B. durch Nutzen von Symmetrie-Eigenschaften des Problems, durch Algorithmen, die mehrere Prozessoren gleichzeitig benutzen können, oder durch ganz andere Ansätze als die oben angegebenen. Vielleicht können Sie auch für jede hinreichend große Zahl n eine „einfache Vorschrift" angeben, mit der man eine bestimmte Lösung des n-Damen-Problems erzeugen kann?

6. Überlegen Sie sich sinnvolle Verallgemeinerungen des Problems, etwa für

- höherdimensionale Schachbretter;
- andere Brett-Geometrien,

  z.B. rechteckige oder torusförmige (d.h. der linke Rand ist mit dem rechten Rand verbunden, und der obere Rand ist mit dem unteren Rand verbunden) Schachbretter;

- Figuren, die auf andere Art ziehen können als die hier betrachteten Damen,

  z.B. „Superdamen“, die wie herkömmliche Damen und zusätzlich wie Springer ziehen können (ein Springer-Zug besteht aus einem Sprung von einem Feld (i, j) auf ein Feld (i', j'), das entweder zwei Felder vertikal und ein Feld horizontal vom Ursprungsfeld entfernt ist, oder zwei Felder horizontal und ein Feld vertikal; die Zugmöglichkeiten einer Superdame sind unten abgebildet).