

Übungen zu Einführung in die Informatik

Freiwillige Abgabe bis zum 19. 11. 2007, 12.00h

Aufgabe 4-1

Damenproblem

Gegeben sei ein schachbrettartiges Spielbrett der Grösse 8 x 8. Auf diesem Brett befindet sich nur eine einzige Sorte von Figuren, die sich wie eine Dame auf einem Schachbrett verhält, d.h. sie kann andere Figuren in derselben Reihe und derselben Spalte sowie auf den Diagonalen, die von ihrem eigenen Feld ausgehen, bedrohen bzw. schlagen. Die Spielfiguren unterscheiden sich nicht nach Farbe, so dass jede Dame jede andere bedroht, die auf einem entsprechenden Feld steht. Eine beliebige Denksportaufgabe besteht darin, eine gegebene Zahl von Damen so auf dem Spielbrett zu plazieren, dass keine Dame bedroht ist.

Dieses Problem soll natürlich nicht in dieser Aufgabe gelöst werden, aber ein bestimmtes Teilproblem, das die Lösungssuche unterstützt, soll programmiert werden.

Laden Sie die Klasse `QueenDemo` von unserer WWW-Seite herunter, und implementieren Sie die Methode `public static boolean[][] queensInDanger(boolean[][] queens)` in der Klasse. Diese Methode testet für jede Dame aus der Ausgangskonfiguration (Spielbrett, `boolean[][] queens`), ob diese von einer anderen Dame bedroht ist. Die Ausgangskonfiguration besteht aus einem 2-Dimensionalen `boolean`-Array, der für jede Dame ein `true` und für jedes leere Feld ein `false` speichert. Es soll ein Spielbrett (2-Dimensionaler `boolean`-Array) zurückgegeben werden, das für jede bedrohte Dame ein `true` und für jede unbedrohte Dame sowie für alle unbesetzten Spielfelder ein `false` angibt. Verwenden Sie zum Testen die Methoden `boardToString(boolean[][] queens)` und `createBoard()`.

Aufgabe 4-2

Josephus

Legenden zufolge (siehe z.B. Wikipedia, http://en.wikipedia.org/wiki/Josephus_problem) wurden in einem Krieg der jüdische Historiker Flavius Josephus und 40 seiner Kamaraden von römischen Soldaten in einer Höhle eingeschlossen. Statt sich zu ergeben beschlossen sich die 41 Männer, sich lieber auf folgende Weise umzubringen: Sie stellten sich in einem Kreis hin und, beginnend vom grössten Mann in ihrer Reihe, ließen zwei Männer (vorerst) am Leben und brachten den dritten um, dann wurde wieder zwei Männer „geschont“ und der dritte ums Leben gebracht. Dieses „Zählen-und-Töten“ sollte im (immer kleiner werdenden) Kreis solange fortgesetzt werden, bis der letzter Mann letztendlich sich selber umbringt. Dabei soll Josephus dank seines mathematischen Talents schnell erkannt haben, welche Position die letzte in der Tötungsreihe ist, und diese auch eingenommen haben, um sich dann den Römern zu ergeben.

Laden Sie die Klasse `Josephus` von unserer WWW-Seite herunter, und vervollständigen Sie die Methode `lastOne`, die zurückgibt, welche Position Josephus bei unsprünglichen n Männern eingenommen haben muss, wenn jeder k -te umgebracht werden soll. Der boolische Array `alive` soll die Männer darstellen: `true` gibt einen lebendigen, `false` einen toten an.