

## Übungen zu Einführung in die Informatik I

### Aufgabe 31      **Zeileneditor (Fortsetzung)**

[siehe Übungsblatt 11]

### Aufgabe 32      **Einfacher Thread**

Schreiben Sie ein Programm, das fünf Threads erzeugt. Die Threads sollen mit einem zufälligen Zeitabstand von ein bis drei Sekunden erzeugt und fortlaufend nummeriert (Thread-Nummer) werden. Alle Threads besitzen einen gemeinsamen Zähler. Jeder der Threads soll nun nach einem zufälligen Zeitintervall zwischen einer und drei Sekunden den gemeinsamen Zähler inkrementieren und zusammen mit seiner Thread-Nummer ausgeben. Jeder Thread soll diesen Vorgang solange wiederholen bis der gemeinsame Zähler 42 erreicht und dann terminieren.

### Aufgabe 33      **Parallele Matrixmultiplikation**

Nutzen Sie das Java Thread-Konzept für die parallele Multiplikation von Matrizen. Auf der WWW-Seite der Übung zu „Einführung in die Informatik I“ finden Sie als Anhang zu Aufgabenblatt 12 eine Java-Implementierung zweidimensionaler Matrizen mit natürlichzahligen Elementen. Sie können die Klasse `Matrix` zur Lösung dieser Aufgabe nutzen. *Wdh: Matrix-Multiplikation  $C_{m \times s} = A_{m \times n} \cdot B_{n \times s}$ ; Element  $c_{i,j} = \sum_{k=1}^n a_{i,k} * b_{k,j}$*

Implementieren Sie eine Klasse `RowColumnMult` als Erweiterung der Klasse `Thread`, deren Instanzen genau ein Element der Ergebnismatrix berechnen.

Dem Konstruktor werden die Eingabematrizen  $A$  und  $B$ , die Ergebnismatrix  $C$  und die Zeile und Spalte des zu berechnenden Elements übergeben. In der `run` Methode wird das gewünschte Ergebniselement berechnet und in die Ergebnismatrix  $C$  geschrieben.

Fügen Sie der Klasse `RowColumnMult` die Methode `public void main(String args[])` hinzu, in dem Sie exemplarisch zwei Matrizen  $A$  und  $B$  erzeugen. Berechnen Sie alle Elemente der Ergebnismatrix  $C$  parallel, indem sie für jedes Ergebniselement einen Thread der Klasse `RowColumnMult` erzeugen und starten. Geben Sie das Multiplikations-Ergebnis auf dem Bildschirm aus.