

Achte bei Deinen Lösungen diesmal besonders darauf, geeignet auf Exceptions zu reagieren!

1 Kommentare entfernen (3+2 Punkte)

Schreibe ein Programm, das aus einer Eingabedatei alle Java-Kommentare entfernt, die mit `//` markiert sind. Die Datei ohne Kommentare soll in eine andere Datei geschrieben werden. Das Programm soll die Namen der Ein- und der Ausgabedatei auf der Kommandozeile entgegennehmen.

Für die Bonuspunkte soll das Programm zusätzlich Kommentare der Form `/* ... */` entfernen. Die Zeichen `//` und `/*` haben innerhalb der jeweils anderen Kommentartypen keine besondere Bedeutung. Du kannst annehmen, dass die Kommentare syntaktisch korrekt und insbesondere nicht geschachtelt sind.

2 Hexdump (3+1 Punkte)

Schreibe ein Programm, das den Inhalt einer Datei Byte für Byte ausgibt. Auf jeder Zeile sollen 16 Bytes als Hexadezimalzahlen ausgegeben werden. (Wenn das Zeichen als `byte` einen negativen Wert b hat, soll $b + 256$ ausgegeben werden.) Wenn die Anzahl der Bytes in der Datei nicht durch 16 teilbar ist, soll die letzte Zeile mit Nullen aufgefüllt werden.

Für den Bonuspunkt soll das Programm nach der Hexadezimal-Darstellung noch die einzelnen Bytes als Zeichen darstellen. Druckbare Zeichen sollen direkt angezeigt werden, während Kontrollzeichen (im Sinne von `Character.isISOControl`) als Punkte gedruckt werden.

Beispiel (Eingaben sind *kursiv* gedruckt):

```
koller@cicero:~$ java Hexdump Hexdump.java
```

```
2F 2F E4 0D 0A 69 6D 70 6F 72 74 20 6A 61 76 61 //?..import java
2E 69 6F 2E 2A 3B 0D 0A 0D 0A 0D 0A 63 6C 61 73 .io.*;.....clas
73 20 48 65 78 64 75 6D 70 20 7B 0D 0A 20 20 20 s Hexdump {...
...
```

Hinweis: Verwende die Klasse `FileInputStream`.

3 Web-Crawler, Teil 1 (3 Punkte)

Schreibe ein Programm, das ein HTML-Dokument von einer URL herunterlädt und die Menge der von dort verlinkten URLs (mit ``-Elementen) aufsammelt. Das Programm soll mit relativen Links umgehen können.

Folgende Klassen könnten nützlich sein:

- `URL` im Package `java.net` stellt eine URL dar und kann einen `InputStream` zurückgeben, von dem der Inhalt der Webseite ausgelesen werden kann. Man kann diesen `InputStream` einem Konstruktor der Klasse `InputStreamReader` übergeben.
- `ParserDelegator` im Package `javax.swing.text.html.parser` und `HTMLToolkit.ParserCallback` im Package `javax.swing.text.html` stellen einen HTML-Parser zur Verfügung. Ein Beispielprogramm, das diese Klassen verwendet, findest Du unter <http://java.sun.com/products/jfc/tsc/articles/bookmarks/>.

Nächste Woche werden wir das Programm zu einem vollständigen Web-Crawler erweitern. Achte also schon mal darauf, dass Deine Klasse geeignete Methoden anbietet, um bequem viele URLs parsen zu können.

4 Exceptions zweckentfremden (1 Punkt)

Wie wir wissen, ist Restrekursion in Java für den Compiler nicht so einfach zu optimieren. Man kann aber Exceptions verwenden, um dennoch sofort nach Berechnung des korrekten Funktionswertes zum ursprünglichen Aufrufer zurückzukehren. (Konstanten Speicherbedarf bekommt man damit natürlich nicht.)

Schreibe eine Version der Fakultätsfunktion, die Exceptions in dieser Art zweckentfremdet. Ist das guter Programmierstil oder nicht?

Abgabe bis 21. 6. 2004, 9 Uhr
(java-uebungen@coli.uni-sb.de)