



LV 4342

Skriptsprachen-Praktikum

Übung 03

Iteratoren
Eine kleine Klasse



Organisatorisches



- Arbeitsverzeichnis:
`~/lv/skriptspr/03/`
- Dateinamen:
`03-lotto.rb` # neu erstellen & abgeben
- Werkzeuge:
`ruby` # Der Interpreter
`irb` # Interactive Ruby-Shell
`emacs, xemacs` # mit Ruby-Mode
`scite` # Ein portabler Editor, auch
 # mit Ruby-Mode
`ri` # Ruby-Dokumentation auf der Kommando-
 # zeile
`komodo` # IDE für Ruby u.a. Skriptsprachen
- Vorlagen:
`(keine)`



Die Aufgabe



- Allgemeine Beschreibung
 - Ziehen Sie Lottozahlen - mittels Iteratoren!
- Material, Hinweise:
 - Zufallszahlen ermitteln Sie mit der Kernel-Methode `rand`
 - Klasse "Array" besitzt eine Methode `sort`
 - Klasse "Fixnum" besitzt einen Iterator `upto`



Die Aufgabe



A: Implementieren Sie eine Klasse "Lotto" mit folgenden Methoden:

```
initialize(limit=49)
    # Bereite die Zahlen 1 .. limit zur Ziehung vor

draw_one                                --> anInteger
    # Ziehe eine Lottozahl aus dem Intervall

draw_all(n=6)                            --> anArray
    # Ziehe n Lottozahlen, liefere sie als Array

draw(n=6) { |num| block }                --> LottoObject # Iterator
    # Ziehe n-mal eine Lottozahl, führe jedes mal
    # block aus, übergebe dabei gezogene Zahl an block.
    # Tipp: Nutzen Sie den Codeblock für die Ausgabe!

draw_sorted(n=6) { |num| block }        --> LottoObject
    # Analog draw, aber aufsteigend sortiert
```



B: Schreiben Sie dazu eine kleine interaktive Anwendung:

Das Ruby-Programm soll den Anwender fragen

- a) nach der Anzahl Kugeln insgesamt (default = 49)
- b) nach der Anzahl zu ziehender Kugeln (default = 6)

Es soll dann in einer Schleife

- a) fragen nach dem Betriebsmodus (s.u.)
- b) eine Lottoziehung im entsprechenden Modus durchführen und das Ergebnis ausgeben, incl. Zusatzzahl (--> "draw_one")

Gewünschte Modi:

- a) Blockweise (alle Zahlen zugleich) und sortiert, mittels Methode "draw_all" und Array.sort
- b) Iteriert, mittels Methode "draw"
- c) Sortiert, mittels Methode "draw_sorted"
- d) (Ende), Schleife verlassen



- Vereinfachungen / Zwischenschritte
 - 1: Arbeiten Sie zunächst mit festen Werten für die Ziehungsparameter (6 bzw. 49). Verallgemeinern Sie, wenn die Kernfunktionen arbeiten.
 - 2: Erlauben Sie zunächst, dass Zahlen mehrfach gezogen werden können. Korrigieren Sie dies, wenn der Rest funktioniert - an zentraler Stelle in "draw_one".
 - 3: Stellen Sie zunächst den Blockmodus fertig, dann erst den Iterator-Modus, zuletzt den Sortier-Modus.
 - 4: Lassen Sie die Schleife über die Modi zunächst aus, testen Sie jeden Modus einzeln im direkten Durchlauf. Implementieren Sie die Schleife erst, wenn alle Modi einzeln funktionieren.
 - 5: **Im Zentrum der Übung** stehen die **Iteratoren**, also die Methoden der Klasse "Lotto". Halten Sie die Benutzerführung bewusst schlicht!



Beispiel-Dialog (voll ausgebaut)



```
$ ./03-lotto.rb
Lottoziehung
Wie viele Zahlen ziehen?: 6
Aus wie vielen?: 49
1=en bloc, 2=Iteriert, 3=Sortiert, q=quit
Ihre Wahl: 1
En bloc: 2 5 27 30 47 49 Zusatzzahl 25
1=en bloc, 2=Iteriert, 3=Sortiert, q=quit
Ihre Wahl: 2
Iteriert: 2 16 1 4 15 3 Zusatzzahl 10
1=en bloc, 2=Iteriert, 3=Sortiert, q=quit
Ihre Wahl: 3
Sortiert: 4 13 31 34 39 42 Zusatzzahl 16
1=en bloc, 2=Iteriert, 3=Sortiert, q=quit
Ihre Wahl: q
$
```