

PS Software Techniken SS 2012

Andreas Naderlinger
andreas.naderlinger@cs.uni-salzburg.at

Allgemeines

- 2-wöchentlich, Montag 08:30 - 10:00
– T03
- Termine: PLUSonline
- Homepage zum PS:

www.softwareresearch.net → Teaching

Allgemeines (2)

- Anwesenheitspflicht!
 - Einmaliges unentschuldigtes Fehlen möglich.
 - In allen anderen Fällen ist eine glaubwürdige Begründung (ev. Attest) *selbständig* vorzulegen.
 - Die ausgearbeitete Lösung ist im Falle eines Fehlens ebenso *unaufgefordert* via E-Mail abzugeben.

Allgemeines (3)

- Aufgaben
 - Praktische Umsetzung
 - Selbständige Erarbeitung von Hintergrundwissen
- 2er Teams
- Präsentation der Ausarbeitungen zu den Aufgaben
 - Theorie, eigene Implementierung
 - Herausarbeiten der Kernaspekte
 - keine Scroll-Down-Demo
 - Gute Aufteilung unter den Teammitgliedern
 - ohne Unterlagen

Allgemeines (4)

- Abgaben
 - andreas.naderlinger@cs.uni-salzburg.at
 - Subject: **PS Software Techniken**
 - .zip-File mit Verzeichnis ,nachname1_nachname2‘
 - Quelldateien (keine Kompilate)
 - Ev. Beschreibung (Überlegungen) als PDF oder TXT
 - Ev. Folien (für Präsentation)
 - Harte Deadlines
- Benotung (individuell)
 - Aufgaben (Anzahl u. Qualität)
 - Präsentation
 - Mitarbeit

INHALT

INHALT: SCALA

Scala

- Programming language
 - by Martin Odersky (ETH, EPFL)
 - started in 2001; 1st release in 2003
- Multi-paradigm
 - Object-oriented (Java, C#, C++)
 - Functional (Haskell, OCaml, F#, ...)

- Runs on Java platform (JVM) (- also version for .NET)
- Scala byte code is nearly identical to Java byte code
- Interoperable with Java and its libraries

A Scalable Language

- „*Scala allows users to grow and adapt the language in the directions they need by defining easy-to-use **libraries that feel like native language support.***“

```
actor {  
  var sum = 0  
  loop {  
    receive {  
      case Data(bytes)          => sum += hash(bytes)  
      case GetSum(requester) => requester ! sum  
    }  
  }  
}
```

- Imagine you have a String variable `name`, and you want to find out whether or not that String contains an upper case character.
- Java

```
boolean nameHasUpperCase = false;
for (int i = 0; i < name.length(); ++i) {
    if (Character.isUpperCase(name.charAt(i))) {
        nameHasUpperCase = true;
        break;
    }
}
```

Example

- Imagine you have a String variable `name`, and you want to find out whether or not that String contains an upper case character.

- **Java**

```
boolean nameHasUpperCase = false;
for (int i = 0; i < name.length(); ++i) {
    if (Character.isUpperCase(name.charAt(i))) {
        nameHasUpperCase = true;
        break;
    }
}
```

- **Scala**

```
val nameHasUpperCase = name.exists(_.isUpper)
```

- Bestehende Java Applikation bzw. Library umschreiben
 - Kompaktheit Fehlerwahrscheinlichkeit
 - Lesbarkeit
 - → Programmieren ohne Seiteneffekte
 - Performance

Assignment 1

- Install Scala
- Hello, world !
 - Script
 - Application (diesmal keine Abgabe per mail nötig)
- <http://www.scala-lang.org>
 - Compiler, Interpreter, Doku, ...
- IDE plug-in (eclipse, ...)
- Online book by Martin Odersky (see PS homepage for link)

2er Teams!