

Proseminar „Programmierung in R“

Julian Kunkel, Jakob Lüttgau, Eugen Betke

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

2015-04-02

1 Einleitung

2 Foliengestaltung

3 Vortragsstil

4 Zitieren

5 Einführung

- Das Proseminar besteht aus mehreren Teilen:
 - Präsentation von 30-35 Minuten Länge
 - + 5 Minuten Diskussion/Fragen + 5 Min Feedback
 - Schriftliche Ausarbeitung
 - Ungefähr 10 Seiten
- Präsentation und Ausarbeitung werden auf unserer Webseite veröffentlicht
- Lesen Sie die organisatorischen Hinweise und tragen Sie sich in die Mailingliste ein!

Zeitplan

- 3 Wochen **vor** der Präsentation
 - Vorlage einer Struktur mit ungefährem Inhalt der Präsentation und Besprechung mit dem Betreuer
- 2 Wochen **vor** der Präsentation
 - Entwurf der Folien, Besprechung des Inhalts und der Form
- 1 Woche **vor** der Präsentation
 - Abgabe der Folien in der finalen Version als PDF (evtl. Druckversion)
- **Vor** Ende des Semesters
 - Abgabe der schriftlichen Ausarbeitung als PDF

- **Hauptsächlich Stichpunkte**
 - Der Grund dafür ist, dass lange Sätze dazu verleiten, diese einfach vorzulesen, was natürlich nicht Ziel einer Präsentation ist. Außerdem lesen Ihre Zuhörer dann auch nur den Text ab und hören Ihnen (zumindest zeitweise) nicht mehr zu.
- „Foliennummer/Folienanzahl“ auf **jeder** Folie
 - Ja, mit PowerPoint ist das nicht so einfach 😊
- Schrift muss gut lesbar sein
 - Z. B. kein Rot auf Schwarz
- Es muss eine Druckversion abgegeben werden
 - Weißer Hintergrund spart Tinte
 - Kann auch ein separates PDF sein
- Animationen nur einsetzen, wenn sinnvoll

Foliengestaltung (2)

- Vermeiden Sie Zweizeiler mit kurzen Anfang, so etwas wie das hier
 - Die Zeilen umschreiben, so dass es Einzeilig wird
 - Oder den Text auf mindestens ein Drittel erweitern
 - Oftmals können auch einfach mehr Stichpunkte entstehen
- Die Folien sollten auch selbstständig verständlich sein
- Einfache Regeln:
 - (Unter)Stichpunkte sollen zum Stichpunkt passen
 - Stichpunkte sollen zum Folientitel passen
 - Folientitel soll zum Kapitel passen
 - Stichpunkte einer Ebene sollten ähnlich von der Struktur sein
 - Die Stichpunkte sollten von oben nach unten lesbar sein

Foliengestaltung (3): Ein negatives Beispiel

- Prozessorarchitektur:
 - Beschreibt die Architektur eines Mikroprozessors
 - Denken Sie an 64 Bit vs. 32 Bit
 - folgt Moores Law
- Mehr über Speichersysteme
 - Es gibt verschiedene Arten:
 - Festplattenbasierte vs. Bandarchive
 - SSDs
 - Man kann weitere Charkateristika unterscheiden.
- Definition: Hochleistungsrechnen
 - Hochleistungsrechnen (englisch high-performance computing, HPC) ist ein Bereich des computergestützten Rechnens. Er umfasst alle Rechenarbeiten, deren Bearbeitung einer hohen Rechenleistung oder Speicherkapazität bedarf.

Foliengestaltung (4): Eine verbesserte Version

- Definition: Hochleistungsrechnen
 - „Hochleistungsrechnen [...] ist ein Bereich des computergestützten Rechnens. Er umfasst alle Rechenarbeiten, deren Bearbeitung einer hohen Rechenleistung oder Speicherkapazität bedarf.“ [Wiki16]
- Prozessorarchitektur
 - Beschreibt den Aufbau eines Mikroprozessors
 - Verfügbare Rechenwerke
 - Speicheransteuerung 64 Bit vs. 32 Bit
- Speichersystem
 - „Speichersysteme sind Systeme für die Online-Daten- verarbeitung sowie zur Ablage, Archivierung und für Sicherungskopien“ [IT16]
 - Speichertechnologie
 - ...
 - Architektur
 - ...

- Oberster Leitspruch: Sie halten den Vortrag für die **Zuhörer!**
- Frei sprechen und nicht einfach nur Folien ablesen
 - Das können die Zuhörer auch selbst 😊
- Halten Sie die Vortragszeit ein
 - Nicht zu kurz, aber auch nicht zu lang
 - Optionale Folien können helfen
 - Schauen Sie regelmäßig auf die Uhr!
- Rechtzeitig und ausreichend vorbereiten
 - **Mindestens** ein Probevortrag
 - Möglichst auch vor anderen Personen
 - Gerne auch bei uns

- Kommen Sie rechtzeitig
 - Beamer ausprobieren (Auflösung, Farben etc.)
 - Tafel wischen
 - Uhr bereitlegen
- Halten Sie den Kontakt zum Publikum
 - Halten Sie Blickkontakt
 - Auf Haltung und Gestik achten
- Kümmern Sie sich um Zwischenfragen
 - Kurze Fragen können Sie direkt beantworten
 - Längere sollten Sie zurückstellen

- „Das Zitat muss einen Zweck erfüllen.“ [Zit14]
- „Du musst das Zitat kennzeichnen und darfst den Inhalt nicht verändern.“ [Zit14]
- „Du darfst nicht mehr zitieren als nötig.“ [Zit14]
- Auch sinngemäße Zitate müssen kenntlich gemacht werden
- Quellen in der Bibliographie angeben



Abbildung:
Karl-Theodor zu
Guttenberg. Quelle:
"Christoph
Braun" [Wiki11]

- Zit14** „Quellen korrekt angeben“,
[http://www.kreisgymnasium-neuenburg.de/
unterricht/itg/quellen-korrekt-angeben](http://www.kreisgymnasium-neuenburg.de/unterricht/itg/quellen-korrekt-angeben), 2014-04-07
- Wiki11** „Karl-Theodor zu Guttenberg, 2011“, [http://de.
wikipedia.org/wiki/Karl-Theodor_zu_Guttenberg](http://de.wikipedia.org/wiki/Karl-Theodor_zu_Guttenberg),
2011-04-07
- IT16** „Speichersystem :: storage system“,
[http://www.itwissen.info/definition/lexikon/
Speichersystem-storage-system.html](http://www.itwissen.info/definition/lexikon/Speichersystem-storage-system.html), 2016-04-06

Introduction to R

- Based on S language for statisticians
- Open source
- Position 19 on TIOBE index
- Interpreter with C modules (packages)
 - Easy installation of packages via CRAN¹
- Popular language for data analytics
- Development tools: RStudio (or any editor), interactive shell
- Recommended plotting library: ggplot2²

Specialties

- Vector/matrix operations. Note: Loops are slow, avoid them
- Table data structure (data frames)

¹Comprehensive R Archive Network

²<http://docs.ggplot2.org/current/>

Course for Learning R Programming

```
1 # Run with "Rscript intro.R" or run "R" and copy&paste
   ↪ into interactive shell
2 # Installing a new package is as easy as:
3 install.packages("swirl")
4 # Note: sometimes packages are not available on all
   ↪ mirrors!
5 library(swirl) # load the package
6
7 help(swirl) # read help about the function swirl
8
9 swirl() # start an interactive course to learn R
10
11 # a simple for loop
12 for (x in 1:10){
13     if (x < 5){
14         print(x)
15     }else{
16         print(x * 2)
17     }
18 }
```

Example R Program

```
1 # create an array
2 x = c(1, 2, 10:12)
3
4 # apply an operator on the full vector and output it
5 print( x*2 ) # prints: 2 4 20 22 24
6
7 # slice arrays
8 print ( x[3:5] ) # prints: 10 11 12
9 print( x[c(1,4,8)] ) # prints: 1 11 NA
10
11 r = runif(100, min=0, max=100) # create array with random numbers
12 m = matrix(r, ncol=4, byrow = TRUE) # create a matrix
13
14 # slice matrix rows "m[row(s), column(s)]"
15 print( m[10:12, ] ) # Output:
16 #           [,1]      [,2]      [,3]      [,4]
17 #[1,] 85.46609 60.749703 10.5062183 7.449173
18 #[2,] 79.76042 52.199321 96.9699856 97.877946
19 #[3,] 37.34286 8.266282 0.3398741 1.957607
20
21 # slice rows & columns
22 print ( m[10, c(1,4)] ) # Output: [1] 85.466085 7.449173
23
24 # subset the table based on a mask
25 set = m[ (m[,1] < 20 & m[,2] > 2) , ]
```