

# Rechnerpraxis

## 07 VIM ✨

Hochschule  
für Technik  
Stuttgart

SS19 | INF 1

Ben Lebherz  
[benjamin.lebherz@hft-stuttgart.de](mailto:benjamin.lebherz@hft-stuttgart.de)

# Vorlesungsplan

Datum	Thema
19. März	Motivation & Organisation
2. April	Grundlagen Unix/Linux
11. April	Dateisystem und Rechteverwaltung
18. April	dotfiles & Linux Installation
25. April	Prozesse und Benutzer
2. Mai	Wiederholung
9. Mai	Shell Magic & Scripting
<b>16. Mai</b>	<b>VIM &amp; <del>Gastvortrag</del></b>
23. Mai	grep, find & Wiederholung
28. Mai	PVL-Vorbereitung
<b>6. Juni</b>	<b>PVL Termin</b>
<i>13. Juni</i>	<i>Pfingstferien</i>
20. Juni	Zusammenfassung & Feedback
27. Juni	Bonus: Wunschthema, Docker, AWS, ...

# Prüfung / PVL

Am **6. Juni 2019**

Anmeldung in **LSF und Moodle**

LSF bis **Montag, 29. April 2019**

Moodle bis **Montag, 20. Mai 2019**

---

## Bewertung

***Bestanden oder Nicht Bestanden - keine Note***

**⚠ Bei Nichtbestehen *keine* Zulassung zur Klausur  
*Einführung in die Informatik* ⚠**

# Prüfung / PVL | 6. Juni 2019

---

## Erlaubte Hilfsmittel

**- jegliche anderen Hilfsmittel werden als Betrugsversuch gewertet -**

- 1x DIN A4 Zettel, doppelseitig, handschriftlich (nicht gedruckt)
- Vorlesungsfolien, ausgedruckt oder am Rechner geöffnet
- Terminal/Shell, am LIDA PC, keine eigenen Rechner

# Prüfung / PVL | 6. Juni 2019

---

- Gruppeneinteilung wird im Moodle-Kurs bekannt gegeben, bitte beachten
- Bei genehmigter, längerer Prüfungszeit bitte rechtzeitig per Mail melden
- Während der PVL darf außer des E-Tests und den Vorlesungsfolien am Rechner kein weiteres Browserfenster / kein weiterer Browsertab geöffnet werden, d.h. es darf nicht auf das Internet zugegriffen werden (dies wird als Betrugsversuch mit entsprechenden Konsequenzen gewertet)
- Digitale Kommunikation zwischen Studenten jeglicher Art ist nicht erlaubt und wird als Betrugsversuch mit entsprechenden Konsequenzen gewertet
- Jegliche Kommunikation zwischen den Gruppen beim Wechsel ist nicht erlaubt und wird als Betrugsversuch mit entsprechenden Konsequenzen gewertet
- Bei technischen Problemen wird der E-Test gegebenenfalls zu einem anderen Termin wiederholt/nachgeholt
- Bei Nichtbestehen der PVL: *Keine Zulassung zur Klausur Einführung in die Informatik*

# Evaluation



<https://evasys-online.hft-stuttgart.de>

20min

VIM 🖊️

# Editoren

- emacs
- nano
- VIM
- gedit
- Kate



# VIM

- VIM steht für „Vlsual“, dem interaktiven Äquivalent zu ed
- VIM steht für „VIM Improved“ und wurde 1988 für den von Bram Moolenaar für den Amiga geschrieben

# VIM - Pro

- Nur mittels Tastatur bedienbar
- Sehr klein & schnell
- Durchgängiges Konzept (reguläre Ausdrücke, etc.)
- Verschiedene Erweiterungen (für C/C++/Java-Programm.)

# VIM - Con

- Nur mittels Tastatur bedienbar
- **Viele** Tastenkombinationen
- Konzept schwierig zu verstehen - **lohnt sich aber**
- Funktionalität idR vorhanden - man muss sie nur kennen/finden

# VIM für Coder

- >200 unterstützte Sprachen
- Syntax-Highlight
- Funktions-Erkennung
- Auto-Einrücken
- Zusammengehörende Klammern

```
$ vim <filename>
```

# VIM Modi

- **Command** Mode (default beim starten)  
"Alle" Tasten werden als Kommando/Funktion interpretiert
- **Insert** Mode  
"Alle" Tasten werden als Text interpretiert
- **Visual** Mode  
Zur Auswahl/Selektion von Textstücken/blöcken

# VIM Modi

- **ESC** für Wechsel in **Command** Mode  
"Alle" Tasten werden als Kommando/Funktion interpretiert
- **i/I, a/A, o/O** für Wechsel in **Insert** Mode  
"Alle" Tasten werden als Text interpretiert
- **v** für Wechsel in **Visual** Mode  
Zur Auswahl/Selektion von Textstücken/blöcken

# VIM Hello World

- `$ vim meine_datei.txt`

Startet VIM und öffnet die Datei `meine_datei.txt`



# VIM Hello World

- **\$ vim meine\_datei.txt**

Startet VIM und öffnet die Datei **meine\_datei.txt**

- **i**

Wechsel in den **Insert** Mode

# VIM Hello World

- **\$ vim meine\_datei.txt**

Startet VIM und öffnet die Datei **meine\_datei.txt**

- **i**

Wechsel in den **Insert** Mode

- Geben sie eine Zeile Text ein

# VIM Hello World

- **\$ vim meine\_datei.txt**

Startet VIM und öffnet die Datei **meine\_datei.txt**

- **i**

Wechsel in den **Insert** Mode

- Geben sie eine Zeile Text ein

- **ESC**

Wechsel zurück in den **Command** Mode

# VIM Hello World

- **:w**  
Speichert **meine\_datei.txt**
- **:x**  
Speichert **meine\_datei.txt** und schließt VIM
- **:q**  
Beendet VIM ohne **meine\_datei.txt** zu speichern  
(ohne Änderungen)
- **:q!**  
Beendet VIM ohne **meine\_datei.txt** zu speichern  
(auch mit Änderungen)

# Nützliche Kommandos

**^** springt zum Zeilenanfang

**\$** springt zum Zeilenende

**:<n>** springt zu Zeile n

**gg** springe zum Dateianfang

**G** springe zum Dateiende

---

**D** löscht bis Zeilenende

**<n>dd** löscht **n** Zeilen / aktuelle wenn kein **n**

---

**<n>yy** kopiert **n** Zeilen / aktuelle wenn kein **n**

**p** fügt Buffer ein

# Nützliche Befehle

**:edit <filename>** öffnet **<filename>** zum editieren

---

**<n>.** wiederholt **n** Befehle / letzten wenn kein **n**

---

**/<Suchwort>** sucht nach  
**n** springt zum nächsten Match für

---

**! <command>** führt **<command>** als Shell-Befehl aus

---

**:%s/<suchwort>/<ersatz>/g** ersetzt **<suchwort>** in  
der ganzen Datei mit **<ersatz>**

# VIM Hands-on! 🧐💻

– Always work in your \$HOME –

---

1. Leiten Sie die Ausgabe von **ps -elf** in eine *neue* Datei **status.txt** um
2. Öffnen Sie VIM und duplizieren Sie die letzten drei Zeilen innerhalb von **status.txt**, speichern Sie die Datei danach erneut
3. Lassen Sie sich die erste Zeile von **ls -lh /usr/bin** anzeigen und leiten Sie diese Zeile in eine Datei **ls.txt** um
4. Öffnen Sie die Datei **ls.txt** mit Vi, kopieren Sie die **Zeilen 3–6** und fügen Sie diese Zeilen **4 mal am Ende des Dokumentes** ein

# VIM Hands-on! 🧐💻

– Always work in your \$HOME –

---

## Kopieren zwischen Dateien

1. Öffnen der Service-Ports Datei **/etc/services** mit VIM
2. Sprung zu Zeile **53**
3. Kopieren Sie **eine Zeile** in den Buffer
4. Editieren Sie eine neue Datei **vi\_test.txt**
5. Fügen Sie den Buffer in diese neue Datei ein
6. Speichern und VIM beenden

## Multifile/line kopieren & einfügen

1. Öffnen der Dateien **/etc/services** und **vi\_test.txt** mit **einem Befehl**
2. Kopieren von **10 Zeilen** aus der ersten Datei
3. **Wechseln** zur zweiten Datei
4. Einfügen dieser **10 Zeilen** am **Ende** der Zweiten Datei



**done!** 🦊

**Thanks!**