

# Übung zu Betriebssysteme

## Aufgabe 7: Eine (graphische) Anwendung, Prüfung, Evaluation & Ausblick

---

04. Februar 2026

Maximilian Ott, Luis Gerhorst, Dustin Nguyen, Phillip Raffeck & Bernhard Heinloth

Lehrstuhl für Informatik 4  
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme  
und Betriebssysteme



Friedrich-Alexander-Universität  
Technische Fakultät

# Ruhmeshalle

via Netboot Ruhmeshalle  
oder QEMU/KVM über /proj/i4bs/halloffame

# Ruhmeshalle

via Netboot Ruhmeshalle  
oder QEMU/KVM über `/proj/i4bs/halloffame`  
(und auch auf der Webseite)

# Ruhmeshalle

via Netboot Ruhmeshalle  
oder QEMU/KVM über </proj/i4bs/halloffame>  
(und auch auf der Webseite)



**teilweise Probleme wegen zu neuer Hardware**

# Anwendung für STUBS (*freiwillige Aufgabe*)

Die Vorgabe enthält

- einen Zufallszahlengenerator
- ein Dateisystem
  - *Minix v3*, von Linux in PASST geklaut
  - mit typischen Schnittstellen
- einen Grafikmodus (VESA)
  - eine *PNG* Bibliothek
- ein kleines Beispiel

# Anwendung für STUBS (*freiwillige Aufgabe*)

Die Vorgabe enthält

- einen Zufallszahlengenerator
- ein Dateisystem
  - *Minix v3*, von Linux in PASST geklaut
  - mit typischen Schnittstellen
- einen Grafikmodus (VESA)
  - eine *PNG* Bibliothek
- ein kleines Beispiel

Ihr braucht noch

- eine dynamische Speicherverwaltung  
(`malloc()` / `free()`, z.B. *Halde* aus SP)

# Anwendung für STUBS (*freiwillige Aufgabe*)

Die Vorgabe enthält

- einen Zufallszahlengenerator
- ein Dateisystem
  - *Minix v3*, von Linux in PASST geklaut
  - mit typischen Schnittstellen
- einen Grafikmodus (VESA)
  - eine *PNG* Bibliothek
- ein kleines Beispiel

Ihr braucht noch

- eine dynamische Speicherverwaltung  
(`malloc()` / `free()`, z.B. *Halde* aus SP)

**Macht was tolles daraus** (wenn ihr wollt & Zeit habt).

# Anwendung für STUBS (*freiwillige Aufgabe*)

Die Vorgabe enthält

- einen Zufallszahlengenerator
- ein Dateisystem
  - *Minix v3*, von Linux in PASST geklaut
  - mit typischen Schnittstellen
- einen Grafikmodus (VESA)
  - eine *PNG* Bibliothek
- ein kleines Beispiel

Ihr braucht noch

- eine dynamische Speicherverwaltung  
(`malloc()` / `free()`, z.B. *Halde* aus SP)

**Macht was tolles daraus** (wenn ihr wollt & Zeit habt).

Und schickt uns das Ergebnis.

# Anwendung für STUBS (*freiwillige Aufgabe*)

Die Vorgabe enthält

- einen Zufallszahlengenerator
- ein Dateisystem
  - *Minix v3*, von Linux in PASST geklaut
  - mit typischen Schnittstellen
- einen Grafikmodus (VESA)
  - eine *PNG* Bibliothek
- ein kleines Beispiel

Ihr braucht noch

- eine dynamische Speicherverwaltung  
(`malloc()` / `free()`, z.B. *Halde* aus SP)

**Macht was tolles daraus** (wenn ihr wollt & Zeit habt).

Und schickt uns das Ergebnis. Irgendwann.

# Prüfung

---

# Prüfung (💀)

---

# Vorbereitung

- Geprüft wird der Stoff der Vorlesung
  - ihr müsst **nicht** den Quellcode (auswendig) kennen
  - aber das Prinzip müsst ihr erklären können!
  - übt mit Kommilitonen, erklärt euch gegenseitig die Vorgehensweise

# Vorbereitung

- Geprüft wird der Stoff der Vorlesung
  - ihr müsst **nicht** den Quellcode (auswendig) kennen
  - aber das Prinzip müsst ihr erklären können!
  - übt mit Kommilitonen, erklärt euch gegenseitig die Vorgehensweise
- alte Prüfungsprotokolle online bei der FSI Informatik:  
<https://fsi.cs.fau.de/dw/pruefungen/hauptstudium/ls4/bs>  
(Schreibt doch nach der Prüfung selbst eines!)

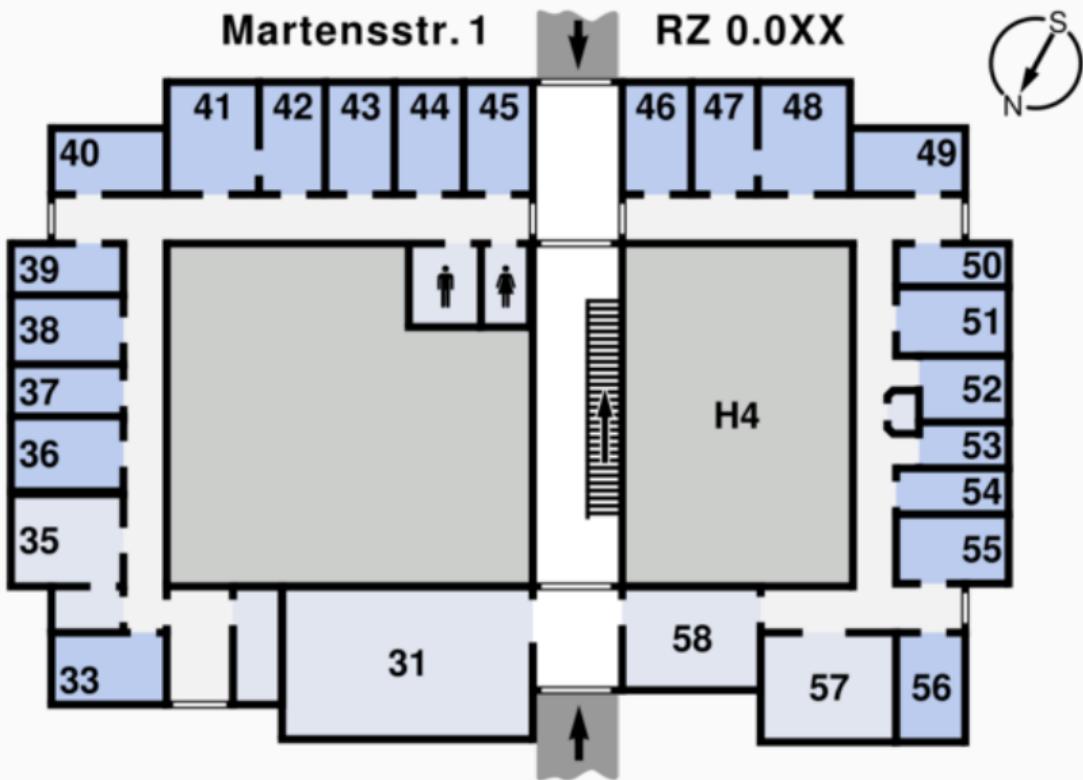
# Vorbereitung

- Geprüft wird der Stoff der Vorlesung
  - ihr müsst **nicht** den Quellcode (auswendig) kennen
  - aber das Prinzip müsst ihr erklären können!
  - übt mit Kommilitonen, erklärt euch gegenseitig die Vorgehensweise
- alte Prüfungsprotokolle online bei der FSI Informatik:  
<https://fsi.cs.fau.de/dw/pruefungen/hauptstudium/ls4/bs>  
(Schreibt doch nach der Prüfung selbst eines!)
- *bei Prüfungsabsage:* Bitte immer eine (kurze) Mail
  - immer. Egal wie kurzfristig.
  - aber je früher desto besser
  - ggf. auch gleich Wunschersatztermin

# Judgement Day: Präsenz

- kommt [über]pünktlich
- und ausgeschlafen ☺
- ein Prüfer und ein protokollierender Beisitzer
- statt schweigend zu denken, lieber eure Überlegung aussprechen
  - man darf nur bepunktten, was ihr von euch gebt  
(und der Prüfer kann euch auf den richtigen Weg bringen)
- sollten Worte fehlen/nicht ausreichen, so habt ihr Stift und Papier
- die 30 Minuten sind schnell vorbei
- ihr bekommt nach weiteren 1-5 Minuten eure Note
- Raum siehe Waffel

# Prüfungsraum



## Ausblick

---

# BST Vorlesung

- Einflussfaktoren
  - I Systemaufruf
  - II Betriebssystemarchitektur
  - III Hierarchien
- Adressraumkonzepte
  - I Seiten: ein-/mehrstufig, invertiert, überwacht
  - II Segmentierung; seitenbasierte Hybride
- Adressraummodelle
  - I Mehradressraumsystem; total/partiell privat
  - II Einadressraumsystem
- Spezialfälle
  - I Adaptiver Speicherschutz (HW/SW; Kohärenz (IPI))
  - II Virtuell gemeinsamer Speicher (PEACE, OctoPOS)
  - III Virtuell nicht-flüchtiger Speicher (PAVE)
  - IV Dynamisches Binden (Multics)

**StuBSmI** – Studentisches Betriebssystem mit Isolation

## **StuBSmI** – Studentisches Betriebssystem mit Isolation

### 1. Privilegientrennung

## **StuBSmI** – Studentisches Betriebssystem mit Isolation

1. Privilegientrennung
2. Systemaufrufschnittstelle

## **StuBSmI** – Studentisches Betriebssystem mit Isolation

1. Privilegientrennung
2. Systemaufrufschnittstelle
3. Paging

## **StuBSmI** – Studentisches Betriebssystem mit Isolation

1. Privilegientrennung
2. Systemaufrufschnittstelle
3. Paging
4. Externe Anwendungen

## **StuBSmI** – Studentisches Betriebssystem mit Isolation

1. Privilegientrennung
2. Systemaufrufschnittstelle
3. Paging
4. Externe Anwendungen
5. Erweiterte Prozess- und Speicherverwaltung

## **StuBSMI** – Studentisches Betriebssystem mit Isolation

1. Privilegientrennung
2. Systemaufrufschnittstelle
3. Paging
4. Externe Anwendungen
5. Erweiterte Prozess- und Speicherverwaltung
6. Nachrichtenaustausch

## **StuBSmI** – Studentisches Betriebssystem mit Isolation

1. Privilegientrennung
2. Systemaufrufsschnittstelle
3. Paging
4. Externe Anwendungen
5. Erweiterte Prozess- und Speicherverwaltung
6. Nachrichtenaustausch
7. Erweitertes Paging

**Viel Erfolg bei der Prüfung  
und schöne Semesterferien!**