

# HEA-Übung

Kira Duwe

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

15-04-2016

## Ein paar Funktionen und Hinweise

- `int open(const char *pathname, int flags)`
- `int open(const char *pathname, int flags, mode_t mode)`
- `write()`: “A successful return from `write()` does not make any guarantee that data has been committed to disk. In fact, on some buggy implementations, it does not even guarantee that space has successfully been reserved for the data. The only way to be sure is to call `fsync(2)` after you are done writing all your data. ”

## Ein paar Funktionen und Hinweise (cont.)

- `fsync()`: “`fsync()` transfers ('flushes') all modified in-core data of (i.e., modified buffer cache pages for) the file referred to by the file descriptor `fd` to the disk device [...] The call blocks until the device reports that the transfer has completed. It also flushes metadata information associated with the file (see `stat(2)`).”
- `close()`: Fehlerbehandlung!! “Not checking the return value when closing the file may lead to silent loss of data. A successful close does not guarantee that the data has been successfully saved to disk, as the kernel defers writes.”

# Fehler und Anmerkungen

- Abgabe muss kompilieren!
- keine Segfaults bei Programmaufruf!
- Fehlerbehandlung wichtig und notwendig für weitere Aufgaben
- Speicherallokation ansehen
- Parameterruf anpassen und usage auch ;)

# iostat und VampirTrace

- fsync einbauen
- iostat für Statistiken
- Spurdaten mit Vampir erzeugen
- Analyse mit VampirTrace
- Messungen auf dem Cluster von verschiedenen Programmversionen