HEA-Übung

Kira Duwe

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

08-04-2016

Anmerkungen

- getattr(): selbstständiges Handhaben der stats
  kein stat(),lstat() etc
- write(): filesize anpassen!
- Fehlerwerte zurückgeben $\neq$ ret 0
- memcpy(): verwenden, nicht selbst bauen

Ergebnisrückgabe in C

- über Rückgabetyp (eher selten)
  `int strcmp(char *str1, char *str2)`
- `int dummyfs_open (const char* path, struct fuse_file_info* fi)`
- Ergebnisrückgabe über Parameter; fd in fi
- Fehlerwert über Rückgabewert; 0 = erfolgreich

Funktionspointer

- int (*POINTER_NAME)(int a, int b)
- z.B: gpointer (*GThreadFunc) (gpointer data)
- http://c.learncodethehardway.org/book/ex18.html

```
1 // typedef creates a fake type, in this
2 // case for a function pointer
3 typedef int (*compare_cb)(int a, int b);
4 int *someFunction(int count,compare_cb cmp)
5 {
6 if(cmp(a,b) > 0) {
7         doSomething();
8     }
9 }
```

Glib

- https://developer.gnome.org/glib/2.48/
- hilfreich für weitere Aufgaben:
  - Basic Types
  - Threads und Mutexe
  - Hash Tables
  - Balanced Binary Trees

Locking

- Signatur: void g_mutex_init (GMutex *mutex);
- Beispiele unter: https://developer.gnome.org/glib/2.48/glib-Threads.html#g-mutex-clear

```
1 GMutex mutex;
2 g_mutex_init (&mutex);
3 g_mutex_lock (&mutex);
4 ...
5 kritischer Codeabschnitt
6 ...
7 g_mutex_unlock(&mutex);
8 g_mutex_clear(&mutex);
```