

# Besprechung der `wsort`<sup>1</sup>

## Tafelübung Systemprogrammierung

Andreas Schärfl <[andreas.schaertl@fau.de](mailto:andreas.schaertl@fau.de)>

SS 2021

---

<sup>1</sup><https://www.schaertl.me/sp1>

Wann und warum fflush?

# Wann und warum `fflush`

- Textausgabe kann überraschend teuer sein<sup>2</sup>.
- Deswegen sind `FILES` so wie auch `stdout` gerne mal gepuffert.

## Wann jetzt `fflush`?

- Wenn die Ausgabe unbedingt vor Ende rausgeschrieben werden soll.
- Bei der `wsort` eigentlich nicht so super wichtig, denn `fflush` passiert automatisch durch `exit`<sup>3</sup>.
- Bitte für SP trotzdem machen, um Fehler bei der Ausgabe in jedem Fall zu erkennen.
- In Netzwerkkommunikation ist `fflush` manchmal allerdings sehr wichtig. Dazu aber mehr nächstes Semester in Systemprogrammierung 2.

---

<sup>2</sup><https://news.ycombinator.com/item?id=10974929>

<sup>3</sup><https://stackoverflow.com/questions/15911517/>

## Gefährliche Strings

# Gefährliche Strings

```
char *mystrdup(const char *str)
{
    size_t len = strlen(str);

    char *buf;

    if ((buf = malloc(len)) == NULL) {
        return NULL;
    }

    strncpy(buf, str, len);
    return buf;
}
```

- Wo ist das Problem?

# Gefährliche Strings

- Wer das 0 am Ende vergisst macht etwas ganz gefährliches.
- Ist `mystring` nicht 0-Terminiert, wird z.B. `puts(mystring)` alle Zeichen bis zum ersten 0 ausgeben
  - → memory dump → private Informationen werden in Netzwerkdiensten publik!

## Wie geht es besser?

- `strncpy` möglichst vermeiden<sup>4</sup>. Auch `strcat` ist fürchterlich.
- `strcpy` ist okay. Zum Kopieren von Strings `strdup` benutzen. Für Formatstrings `snprintf` angucken.
- *Für Interessierte*: Das Projekt Git geht sogar noch weiter und verbietet noch mehr<sup>5</sup>.

---

<sup>4</sup><https://randomascii.wordpress.com/2013/04/03/stop-using-strncpy-already/>

<sup>5</sup><https://github.com/git/git/blob/master/banned.h>

Coole Typen

# Cooler Typen

- Es gibt mehr als nur einen schönen `int`.
- Besonders Sinnvoll: `size_t` ist ein `unsigned` Typ für Größenangaben jeder Art.
  - Funktionen wie `strlen` oder `strcpy` arbeiten alle mit Typ `size_t`.



Besprechung

- Jetzt Besprechung einer beispielhaften Aufgabe.
- Be excellent to each other.

Fragen?