

OPERATING SYSTEMS BEISPIEL 2

Aufgabenstellung – websh

Schreiben Sie ein Programm, welches beliebige Programme aufruft und deren Ausgaben für das Web formatiert.

SYNOPSIS:

```
websh [-e] [-h] [-s WORD:TAG]
```

Anleitung

Der Elternprozess liest bis zum *EOF* Zeilen ein und startet jeweils zwei Kindprozesse. Der erste führt das in der Zeile angegebene Programm inklusive Argumente durch ein beliebiges Mitglied der *exec*-Familie aus. Die Standardausgabe soll dabei an den zweiten Prozess umgeleitet werden.

Der zweite Kindprozess soll also die Ausgabe des ersten bekommen und diese verarbeiten, um das Ergebnis in einem Webbrowser darstellbar auf der Standardausgabe auszugeben. Primär sind daher Zeilennumbrüche zu behandeln. Mit dem Parameter **-e** soll die gesamte Ausgabe in ein vollständiges HTML-Dokument eingebettet werden, also mit umschließenden **html**- und **body**-Tags. Die Option **-h** fügt den Programmaufruf als Überschrift zwischen den jeweiligen Programmausgaben ein, und **-s** sucht nach Zeilen mit **WORD** und setzt sie in den angegebenen **TAG**.

Sie dürfen eine sinnvolle Länge der Zeilen annehmen, als Konstante definieren und damit arbeiten.

Ein Aufruf mit folgenden Eingaben:

```
$ ./websh -e -h -s Linux:h2
uname -sm
ls /dev/null
```

sollte folgende Ausgabe produzieren:

```
<html><head></head><body>
<h1>uname -sm</h1>
<h2>Linux x86_64</h2><br />
<h1>ls /dev/null</h1>
/dev/null<br />
</body></html>
```

Richtlinien

Beachten Sie unbedingt auch die *Richtlinien für die Erstellung von C-Programmen* sowie die *Allgemeinen Hinweise zur Beispielgruppe 2* auf der Übungswebsite!

Insbesondere ist es ab dieser Beispielgruppe notwendig, die Dokumentation in Doxygen zu führen. Eine kurze Einführung haben wir Ihnen im OSUE-Wiki bereitgestellt. Es muss zumindest das HTML-Output generierbar sein. Bitte dokumentieren Sie ausnahmslos alle Funktionen (auch **static**-Funktionen; siehe **EXTRACT_STATIC** in der *Doxyfile*). Achten Sie weiters darauf, dass nach außen hin sichtbare Funktionen

(exportierte Funktionen) in der Header-Datei und lokale (**static**) Funktionen nur in der C-Datei dokumentiert werden. Sie sollten auch Ihre Typen (insbesondere **structs**), Konstanten und globale Variablen dokumentieren.