

2. Assignment

Rudimentary shell implementation

Issue: 04.05.—Due: 11.05.

Exercise 4: *Extend the basic osh implementation*

10 Points

During the lecture we have discussed a very simplistic shell example. The URL ¹ references another rudimentary shell stub you may use to integrate the following exercises into.

Built-in zombie command. Extend the shell stub by an internal command `zombie`. On each invocation `zombie` should create a new zombie process (showing up in the process status list produced by `ps(1)`). Do zombies pose a problem? How can you get rid of them?

Launch executables and display their exit code. Implement a routine to launch arbitrary applications. For example: `[osh] >> xclock -digital` executed inside the shell stub does the same as if it would be invoked on the system's shell, i.e., start up a digital clock. Once the launched program has terminated the exit code can be displayed using the internal command `status` (which you provided).

As usual.

- Prepare an archive (`zip`, `tar`, ...) of all files that contribute to your solution.
- Make sure to only use open file formats, such as `txt`, `pdf`, `png`, `jpg` and the like and please have an eye on moderate file sizes.
- The archive should not contain compiled (executable or object) files.
- Name the archive according to, e.g., `os_ass2_groupN.tar`
- Send the archive containing your solution as an attachment
To: `alexander.holupirek@uni-konstanz.de`, `stefan.klinger@uni-konstanz.de`
Subject: `[OS] Assignment 2 - Group N`

Have fun.

¹<http://www.inf.uni-konstanz.de/dbis/teaching/ss09/os/osh.c>
<http://www.inf.uni-konstanz.de/dbis/teaching/ss09/os/misc.c>
<http://www.inf.uni-konstanz.de/dbis/teaching/ss09/os/misc.h>
\$ cc -Wall osh.c misc.c