

Lab0 Utilities 实习说明

本 lab 的任务是熟悉 xv6 及其系统调用。

详细要求及提示见链接: (<https://pdos.csail.mit.edu/6.828/2020/labs/util.html>)

实习内容

Exercise 0 源代码阅读

阅读下列源代码, 理解 xv6 及其系统调用。

- `kernel/syscall.h`, `kernel/sysproc.c`

Exercise 1 运行 xv6

xv6 环境安装可以参考官方文档 [xv6 tools](#) 或我们整理的中文资料 [xv6-pku-hints](#)。

安装好后, 用 git 克隆 xv6 仓库:

```
$ git clone git://g.csail.mit.edu/xv6-labs-2020
Cloning into 'xv6-labs-2020' ...
...
$ cd xv6-labs-2020
$ git checkout util
Branch 'util' set up to track remote branch 'util' from 'origin'.
Switched to a new branch 'util'
```

运行 xv6:

```
$ make qemu
```

退出 qemu: 按下 `Ctrl-a+x`。(先同时按下 `Ctrl` 和 `a`, 松开后再按 `x`)

自我评分:

```
$ make grade
```

Exercise 2 sleep

为 xv6 实现 UNIX 程序 `sleep`; `sleep` 应暂停用户指定的 tick 数。tick 是 xv6 内核定义的时间概念, 即定时器芯片的两次中断之间的时间。请在文件 `user/sleep.c` 中实现。

Exercise 3 pingpong

实现程序 `pingpong`，该程序使用 UNIX 系统调用在两个进程之间通过管道传递一个字节。

父进程将一个字节发送给子进程；子进程打印 `<pid>: received ping`，其中 `<pid>` 是其进程 ID；子进程将字节发送给父进程，然后退出；父进程从子进程读取字节，打印 `<pid>: received pong`，然后退出。

请在文件 `user/pingpong.c` 中实现。

Exercise 4 primes

使用管道编写并发版本的素数筛。这个想法归功于 Unix 管道的发明者 Doug McIlroy，细节见[链接](#)。

请在文件 `user/primes.c` 中实现。

Exercise 5 find

编写 UNIX `find` 程序的简单版本：查找具有特定名称的目录树中的所有文件。

请在文件 `user/find.c` 中实现。

Exercise 6 xargs

编写 UNIX `xargs` 程序的简单版本：从标准输入中读取多行并为每行运行一个命令，并将该行作为命令的参数。

请在文件 `user/xargs.c` 中实现。