

Lab2 Page Tables 实习说明

本 lab 的任务是理解 xv6 页表的实现。

详细要求及提示见链接：

(<https://pdos.csail.mit.edu/6.828/2020/labs/pgtbl.html>)

实习内容

Exercise 0 源代码阅读

阅读下列源代码。

- `kernel/memlayout.h`
- `kernel/vm.c`
- `kernel/kalloc.c`

Exercise 1 Print a page table

定义一个名为 `vmprint()` 的函数。它接受一个 `pagetable_t` 参数，并按要求的格式打印该页表（详细要求见[链接](#)）。在 `kernel/exec.c` 的 `return argc` 之前，插入 `if(p->pid==1) vmprint(p->pagetable)`，以打印第一个进程的页表。

Exercise 2 A kernel page table per process

修改内核，以使每个进程在内核中执行时都使用其自己的内核页表副本。修改 `struct proc` 来为每个进程维护一个内核页表，并修改调度器以在切换进程时切换内核页表。

Exercise 3 Simplify `copyin/copyinstr`

将 `kernel/vm.c` 中 `copyin` 的函数体修改为对 `copyin_new`（在 `kernel/vmcopyin.c` 中定义）的调用；对 `copyinstr` 和 `copyinstr_new` 做相同的修改。将用户地址的映射添加到每个进程的内核页表，以使 `copyin_new` 和 `copyinstr_new` 起作用。