

# Lab4 Lazy Page Allocation 实习说明

---

本 lab 的任务是实现惰性页表分配。

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

## 实习内容

---

### Exercise 0 源代码阅读

阅读下列源代码。

- `kernel/trap.c`
- `kernel/vm.c`
- `kernel/sysproc.c`

### Exercise 1 Eliminate allocation from `sbrk()`

在 `sbrk(n)` 系统调用实现中删除页面分配, 即 `sysproc.c` 中的 `sys_sbrk()` 函数。  
`sbrk(n)` 系统调用将进程的内存大小增加 `n` 个字节, 然后返回新分配的区域的开始 (即旧大小)。新 `sbrk(n)` 应该只将进程的大小 `myproc()->sz` 增加 `n` 并返回旧的大小。

### Exercise 2 Lazy allocation

修改 `trap.c` 中的代码, 通过映射新分配的物理内存页到故障地址来响应用户空间中的缺页错误, 然后返回到用户空间继续执行进程。应该在调用 `printf` 产生 "usertrap(): ..." 消息前添加代码。修改任何其他需要的 `xv6` 内核代码, 以使 `echo hi` 起作用。

### Exercise 3 Lazytests and Usertests

修改内核代码, 以便 `lazytests` 和 `usertests` 都能通过。