

# گزارش پروژه دوم

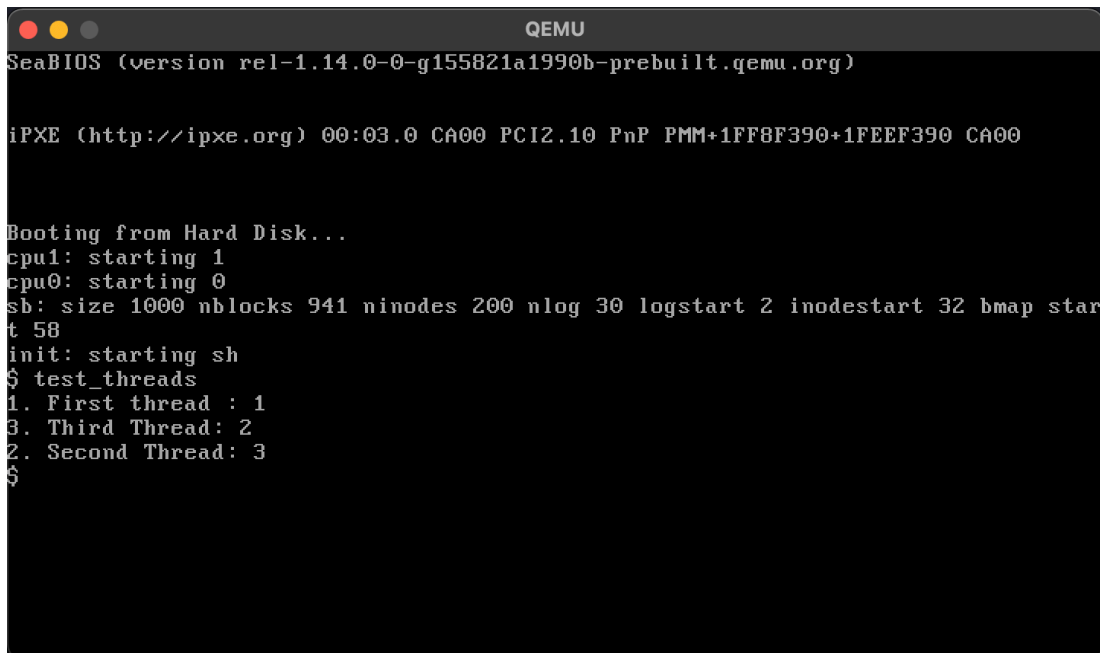
xv6

نیکی نزاکنی  
۹۸۵۲۲۰۹۴

برای تعریف دو سیستم کال جدید به نام های clone و join ابتدا باید در فایل syscall.h آن ها را اضافه کرده و یک شماره به آن ها اختصاص دهیم (۲۳ و ۲۴). همچنین در فایل syscall.c آن ها را تعریف کرده و در فایل sysproc.c خود سیستم کال ها را پیاده سازی میکنیم. برای دسترسی یوزر به این سیستم کال ها باید در فایل usys.s به صورت SYSCALL(clone) و SYSCALL(join) و همچنین در user.h به همراه ورودی هایشان نیز آن را تعریف کنیم. همچنین تابع هایی که قرار است به عنوان library جدید مورد استفاده قرار بگیرند از جمله thread\_create، thread\_join، lock\_init، lock\_acquire و lock\_release به همراه ورودی هایشان باید در فایل user.h تعریف شوند. برای استفاده از مفهوم lock و استفاده آن در فایل user.h یک struct به نام lock\_t تعریف میکنیم که با استفاده از متغیر flag عمل lock را انجام می دهد. در پیاده سازی در فایل sysproc.c، با استفاده از argint ورودی های پاس داده شده به تابع ها را دریافت میکنیم. فایل proc.c جایی است که پیاده سازی دو تابع clone و join انجام شده است به این صورت که تابع clone با اطلاعات stack داده شده به آن یک پروسه جدید میسازد و تابع join در process table به دنبال پروسه های فرزندی میگردد که به حالت zombie در آمده اند تا آن ها را از جدول پاک کند. این دو تابع را در defs.h تعریف میکنیم. همچنین در فایل proc.h که struct proc قرار دارد، داخل این struct یک property جدید به نام threadstack اضافه میکنیم که بتوانیم آدرس thread stack را نگه داریم. Definition تابع های جدید که برای library جدید در نظر گرفتیم باید در فایل ulib.c قرار بگیرند یعنی تابع های thread\_create، thread\_join، lock\_init، lock\_acquire و lock\_release در ulib.c تعریف شده اند. برای تست کردن یک فایل جدید به نام test\_threads.c میسازیم که در آن کاربر با صدا کردن test\_threads، سه thread با استفاده از thread\_create ساخته شده که هر کدام در function های جدا به مقدار counter که در ابتدا صفر است یک واحد اضافه میکنند که چون از lock و function هایی که تعریف کردیم استفاده میکنند thread safety برقرار است و

مطمئن هستیم که counter در آخر مقدار ۳ را خواهد داشت. در آخر این فایل تست را به Makefile اضافه میکنیم که با make کردن این فایل تست هم کامپایل شود.

خروجی:



```
QEMU
SeaBIOS (version rel-1.14.0-0-g155821a1990b-prebuilt.qemu.org)

iPXE (http://ipxe.org) 00:03.0 CA00 PCI2.10 PnP PMM+1FF8F390+1FEEF390 CA00

Booting from Hard Disk...
cpu1: starting 1
cpu0: starting 0
sb: size 1000 nblocks 941 ninodes 200 nlog 30 logstart 2 inodestart 32 bmap start 58
init: starting sh
$ test_threads
1. First thread : 1
3. Third Thread: 2
2. Second Thread: 3
$
```

\*\*\*توضیحات: فایل Makefile برای قابل اجرا بودن روی MacOS تغییر کرده است. در صورت اجرا نشدن این فایل روی Linux، فایل Makefile قبلی با اضافه شدن test\_threads.c نیز در فولدر Makefile\_Linux قرار داده شده است.