



**Department of  
Computer Engineering**

**Project Phase3**

**Fall 2023**

**Dr. Javadi**

## فاز سوم

### ● Multi-level queue scheduling:

ابتدا نحوه ی کارکرد تابع scheduler در فایل proc.c را بررسی کنید و سپس به بررسی توابع مربوط دیگر مانند sched و yield بپردازید و پیدا کنید که چگونه و کجا timer interrupt ایجاد میشود و handle میشود. در بررسی خود به نحوه کارکرد توابع و موارد استفاده و جایی که صدا می شوند بپردازید. سپس بررسی کنید که زمانبند پیش فرض xv6 از چه نوع است و چگونه کار می کند. در ادامه مشخص کنید که چگونه یک پردازش به پایان می رسد و از کدام توابع برای این بخش استفاده میشود. در آخر روش کارکرد یک زمانبند چند صفی را مطالعه کنید.

**دقت کنید موارد مورد بررسی در هنگام تحویل پروژه مورد سوال قرار میگیرند.**

در این بخش قرار است تا زمانبند پیش فرض را تغییر داده و یک زمانبند چند صفی پیاده سازی کنید.

#### ● cpu:

در این بخش لازم است برای درک بهتر کارکرد پیاده سازی خود تعداد پردازنده ها را به ۱ تغییر دهید. برای این کار به فایل Makefile مراجعه کنید.

### ● Multi-level queue:

در این بخش که اکثر پیاده سازی این فاز را شامل میشود، باید ابتدا یک مکانیزم پیاده سازی کنید که بتواند از الگوریتم چند صفی پشتیبانی کند. شما لازم دارید که صف پردازش ها را طوری تعریف کنید که پردازش ها بتوانند در ۳ صف ذخیره شوند. سپس هر جا که لازم شد استراکت proc را طوری تغییر دهید که اگر لازم است اطلاعات مربوط به صفی که پردازش داخل آن هست را ذخیره کند.

هر صف از الگوریتم Round Robin استفاده میکند. صفی که بالا تر است دارای اولویت بیشتری است. یعنی برای مثال صف شماره یک اولویت بیشتری نسبت به صف شماره دو دارد. صف اول دارای کوانتوم زمانی ۵ و صف دوم دارای کوانتوم زمانی ۱۰، و صف سوم دارای کوانتوم زمانی ۲۰ است. پردازش ها در ابتدای ساختن وارد صف اول می شوند.

● در این بخش لازم است تا اگر فیلدی به استراکت proc اضافه کردید آن را در تابع مناسب مقداردهی اولیه کنید.

## ● Multi-level queue scheduler:

در این بخش لازم است تا تابع scheduler را که قبلا بررسی کرده اید طوری تغییر دهید که بصورت چند صفی عمل کند و اولویت صف ها رعایت شوند.

## ● Test:

در این بخش باید یک فایل تست در بخش user آماده کنید و ۱۰ پردازش در آن ایجاد کنید که هر کدام از ۱ تا ۱۰۰۰۰۰۰۰۰۰ شمارش میکند. سپس در پایان اجرا یک پیام چاپ کند که پردازش با شماره ی n (بین یک تا ده) تمام شد.  
همچنین لازم است در هر بار اجرای پردازش که cpu به آن اختصاص داده میشود یک پیام مانند پیام قبل با متن مناسب چاپ کنید و هنگامی که از روی cpu برداشته شده و به صف منتقل میشود نیز یک پیام مناسب چاپ کنید. دقت کنید که در متن پیام ها از pid استفاده کنید. پیام هایی که هنگام اجرا و خروج توسط scheduler چاپ می کنید حتما حاوی اطلاعاتی که پردازش در کدام صف است نیز باشد.

\* در پیاده سازی همه ی بخش ها محدودیتی وجود ندارد و در ایجاد تغییرات می توانید آزادانه عمل کنید.

## بخش امتیازی:

در این بخش لازم است پیاده سازی خود را بگونه ای تغییر دهید که بتوانید برای هر پردازش در ابتدای ساخت آن یک اولویت مشخص کنید و در هر صف زمانبند، الگوریتم زمانبند با توجه به اولویت پردازش ها عمل کند و اگر اولویت ها برابر بود به صورت Round Robin عمل کند.

## توضیحات

- پروژه شما تحویل اسکایی خواهد داشت بنابراین از استفاده از کدهای یکدیگر یا کدهای موجود در وب که قادر به توضیح دادن عملکرد آنها نیستید، بپرهیزید.
- ابهامات خود را در گروه درس در تلگرام مطرح کنید و ما در سریعترین زمان ممکن به آنها پاسخ خواهیم داد.

## آنچه که باید ارسال کنید:

یک فایل زیپ با نام OS\_P3\_Sid.zip (که Sid را با شماره دانشجویی خود جایگزین کنید) که شامل دو مورد زیر است:

- پوشه ای که در آن کدهای شما وجود دارد. دقت کنید که **تنها و تنها فایل هایی را که تغییر داده اید یا اضافه کرده اید** را برای ما بفرستید.

موفق باشید

تیم تدریس یاری درس سیستم های عامل