



## ارزیابی برش زمانی پویا در زمان بند XV6

### مقدمه

برنامه زمان بندی یکی از اجزای بسیار حیاتی در سیستم‌های عامل است که مسئول مدیریت و تخصیص منابع پردازشی به فرآیندها است. در سیستم عامل XV6 که یک سیستم عامل آموزشی و ساده برای ارتباط با سیستم‌های عامل بسیار پیچیده‌تر است، الگوریتم زمان بندی اصلی ثابت و مبتنی بر RR است. این الگوریتم، برش زمانی‌های ثابتی را به هر فرآیند بدون توجه به نیازهای واقعی آن اختصاص می‌دهد. در این پروژه، ما قصد داریم تا با اضافه کردن یک برش زمانی پویا به برنامه زمان بندی XV6، به بهبود عملکرد سیستم بپردازیم. با برش زمانی پویا، سیستم عامل قادر خواهد بود تا برش زمانی‌های مختلفی را به فرآیندها اختصاص دهد، که با توجه به نیازهای آن‌ها و شرایط سیستم، تنظیم می‌شوند. این بهینه‌سازی می‌تواند باعث افزایش عملکرد و بهبود واکنش پذیری سیستم در مقابل بار کاری متغیر شود.

### پیاده‌سازی

همانطور که گفته شد زمان بند این سیستم عامل آزمایشی RoundRobin است.

#### فعالیت

به تابع schedule رفته و نحوه کارکد مربوطه را توضیح دهید. ساختار ptable و جزییات داخل تابع که برای شما ناشناخته است را تحقیق کرده و توضیح دهید.

### تغییرات زمان بند

ابتدا به بررسی الگوریتم زمان بندی اصلی در XV6 می‌پردازیم و سپس مراحل اضافه کردن برش زمانی پویا را به برنامه زمان بندی معرفی می‌کنیم. سپس اثرات این تغییرات بر عملکرد سیستم را مورد ارزیابی و بررسی قرار می‌دهیم. تصویر زیر نحوه کارکرد عادی الگوریتم فعلی سیستم را نشان می‌دهد. در محور افقی هر خانه یک Time Slice یا برش زمانی است. در محور عمودی نیز ما برنامه‌های سطح کاربر را داریم.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
proc0	TS			TS			TS			TS			TS		
proc1		TS			TS			TS			TS			TS	
proc2			TS			TS			TS			TS			TS

\* exit  
Round Robin

#### فعالیت

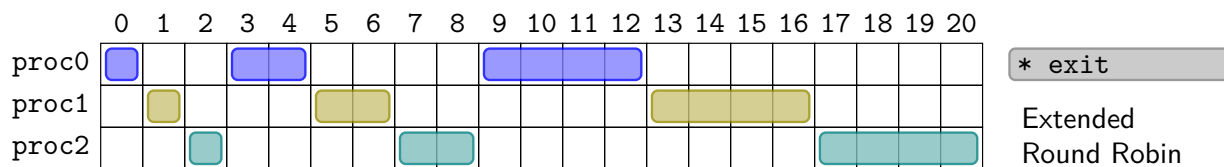
در چه صورت زمان بند اجرا می‌شود؟ درباره نحوه قبضه در این سیستم عامل تحقیق کنید. نحوه کار کلی سخت افزار مربوط به Timer را به طور خلاصه توضیح داده و سعی کنید محل تنظیم این سخت افزار را پیدا کنید.

#### راهنمایی

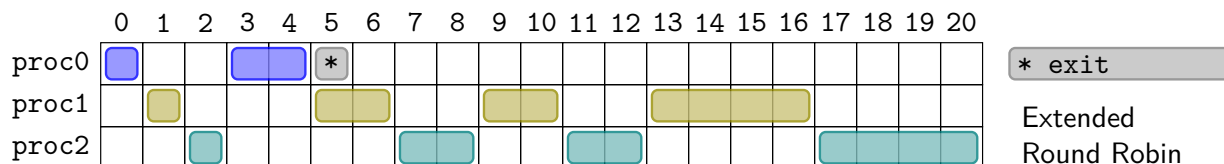
برای انجام این فعالیت می‌توانید درباره توابعی که تابع schedule را صدا می‌زنند با استفاده از جستجوگر ویرایشگر خود تحقیق کنید. این کار را تا جایی ادامه دهید که تمام Stack‌های ممکنه که در آن زمان بند صدا زده می‌شود را بدست آورید. سپس شروع به تحقیق درباره عملکرد کلی هر تابع موجود در استک کرنل شوید.

سناریو های متفاوت شامل خروج برنامه، TimerInterrupt و Wait را بدست آورده باشید. در نهایت برای Timer می‌توانید در تابع main به دنبال مسیر اجرای تابع lapic (Local Advanced PIC) تحقیق کنید.

می‌خواهیم قابلیت به این زمان بند اضافه کنیم که در صورت تمام نشدن هیچ فرآیندی در یک دور کارکرد این زمان بند، طول برش زمانی تخصیصی در دوره دوبرابر شود. مثال:



همچنین اگر در یک دور فعالیت زمان بند، یکی از فرآیندها به اتمام کار خود رسید، طول برش زمانی ثابت بماند. به عنوان مثال در تصویر زیر فرآیند صفر بعد از برش زمانی دوم کارش تمام شده و مانع افزایش طول بازه می شود.



#### فعالیت

طول هر بازه زمانی را چگونه افزایش می دهیم؟ طول برش زمانی مگر یک واحد پیوسته بر حسب ثانیه نیست؟ سیستم عامل بر چه حساب این زمان را حساب می کند و چه مفهومی دارد؟

#### راهنمایی

بعد از انجام فعالیت بالا به تمام جزئیات مربوط به Context Switch پی برده باشید. در این مسیر احتمالا به تابع trap رسیده اید. درباره Tick تحقیق کنید. سعی کنید با اضافه کردن یک متغیر شمارنده به سیستم این کارکردها را به سیستم اضافه کنید.

## ارزیابی تغییرات

ابتدا سعی کنید برنامه هایی در سطح کاربر ایجاد کنید که Load زیاد و لازمه روی سیستم ایجاد شود و ما تعداد Context Switch های کافی را شاهد باشیم. معیارهای درس داده شده در سرکلاس برای ارزیابی عملکرد یک زمان بند را به یاد بیاورید. در سیستم مکانیزم هایی برای اندازه گیری این معیارها اضافه کنید. به عنوان مثال سعی کنید سیستم کال هایی را بسازید که به شما زمان بازگشت را تحویل دهد یا زمان خروج فرآیند از cprintf استفاده کند. یا سیستم کالی که تعداد Context Switch های انجام شده را نیز در انتهای آخرین برنامه از زمان شروع آن ها را روی صفحه چاپ کند.

#### فعالیت

تحلیل هایی روی این معیارها قبل و بعد از اجرای تغییرات ارائه دهید. توضیح دهید برنامه های شما چه تاثیری روی این معیارها دارند. گزارشی از معیارها و نحوه تغییر آن ها تحت تست های مختلف بنویسید.

#### راهنمایی

یک برنامه ضرب ماتریس و حلقه های تودرتو می تواند بار محاسباتی خوبی به سیستم تحمیل کند. همچنین سعی کنید یک برنامه دیگر با استفاده از ورودی و خروجی و IO را بکار بگیرید تا تست ها واقعی تر شوند. استفاده از دیسک در این مورد خوب است. خواندن کد برنامه های مشابه نمونه موجود در کامند ls، می تواند به شما توانایی تولید موارد مشابه و پیچیده تر را بدهد. استفاده از fork برای ایجاد ترکیب های مختلف از دو برنامه بالا را از یاد مبرید.

## نحوه ارسال

قبل از شروع پروژه، یک مخزن خصوصی برای خود ساخته و هم گروهی خود را به عنوان مشارکت کننده به آن اضافه کنید. سپس در اولین کامیت مخزن خود کد فعلی XV6 را قرار داده و با نام Initial XV6 آن را کامیت کنید. برای ارسال کار خود، به سادگی تمام تغییرات خود را (و فایل های جدید در صورت نیاز) کامیت کرده و کار خود را به مخزن GitHub خود بفرستید. همچنین، یک فایل فشرده از مستندات فارسی و مخزن را در Quera بارگذاری کنید. اطمینان حاصل کنید که نام فایل را با شماره دانشجویی های خود امضا کنید. اگر این کار را انجام ندهید، قادر به مرتبط سازی ارسال شما با شما نخواهیم بود!

برخی نکات قابل توجه:

- **پیشرفت تدریجی:** تمرکز بر روی توسعه‌ی تدریجی برای حفظ قابلیت کامپایل است. هر عملکرد اضافه شده را به صورت جداگانه اشکال‌زدایی کنید.
- **مشارکت فعال:** همه اعضای تیم باید به مشارکت فعال در پروژه بپردازند. برای ارائه نمودار همکاری در روز ارائه پروژه، افزونه‌های Gitgraph یا GitLens را برای VSCode نصب کنید. اعضای غیرفعال تیم ممکن است نمره پایین‌تری دریافت کنند.
- **ایجاد یک مخزن خصوصی:** مخزن خود را تا روز ارائه به صورت خصوصی نگه دارید. اطمینان حاصل کنید که کامیت اولیه شما آخرین کامیت از مخزن xv6-public است. به ما این امکان را می‌دهد تا از دستور Diff در لینوکس برای ردیابی تغییرات استفاده کنیم.
- **تست:** هر فراخوانی سیستم را به صورت جداگانه یا آن‌ها را در سناریوهای ترکیبی و مختلف تست کنید. عملکردهای تست‌نشده نمره‌دهی نخواهند شد.
- **مستندسازی چالش‌ها:** مستندات جامعی راجع به چالش‌هایی که در طول پروژه با آن‌ها روبه‌رو شده‌اید، به همراه رویکردهای حل مشکل بنویسید. بخش‌هایی از پروژه که به صورت دقیق تعریف نشده‌اند را شرح دهید و نحوه مدیریت آن‌ها را بیان کنید. ما مشتاق هستیم تا فرآیند اندیشیدن شما و راه حل‌هایی که پیاده‌سازی کرده‌اید را درک کنیم.
- **آمادگی ارسال:** یک بررسی کامل کامپایل قبل از ارسال انجام دهید. کدی که قابل کامپایل نیست نمره‌دهی نخواهد شد.

#### هشدار

شما باید به قدر به بازتولید پروژه انجام‌شده در زمان ارائه باشید. علاوه بر تسلط بر کد زده‌شده، باید به علت تغییرات ایجاد شده و مفاهیم درخواست‌شده و مورد نیاز برای انجام پروژه کامل تسلط داشته باشید تا به سوالات دستیاران حل تمرین پاسخ دهید. استفاده از منابع داخل اینترنت تنها در صورت تسلط کامل بلامانع است. در غیر این صورت به علت عدم صداقت در انجام پروژه، می‌تواند به اخذ تقلب و ایجاد مشکل در فرآیند درسی دانشجوی یا کسب نمره صفر در پروژه منجر شود.