دانشکده مهندسی کامپیوتر

درس سیستم عامل دکتر رضا انتظاری ملکی

فاز اول پروژه

طراح ..... مرتضی ملکی نژاد شوشتری

تاریخ انتشار ..... ۲۷ آبان ۱۴۰۳

تاریخ تحویل ..... ۲۳ آذر ۱۴۰۳

۱ توضیحات

۱.۱ مقدمه

xv۶ یک سیستم عامل آموزشی است که در درس سیستم عامل از آن استفاده میشود. شما در این فاز از پروژه باید بر روی سیستم عامل xv۶ قابلیت تعیین بیشینه استفاده از منابع (CPU و Ram) را برای پروسه ها اضافه کنید. به این صورت که اگر پروسه ای از میزان تعیین شده، CPU بیشتری نیاز داشت، نباید بتواند بگیرد و اگر مصرف مموری آن از حد تعیین شده فراتر رفت، دیگر نتواند حافظه ای بگیرد.

۲.۱ CPU

برای این قسمت، شما باید کدی بزنید که اجازه ندهد در یک ثانیه یک پروسه بر روی پردازنده بیش از درصد مشخصی در حال اجرا باشد. برای سادگی حالت تک هسته ای را در نظر بگیرید و فرض کنید که ثانیه ها مجزا از هم هستند (یعنی اگر یک پروسه محدودیت ۵۰ درصد داشته باشد، ممکن است که ۵۰۰ میلی ثانیه آخر ثانیه اول و ۵۰۰ میلی ثانیه اول ثانیه دوم در حال اجرا باشد.)

در واقع شما باید یک syscall برای تعیین محدودیت بنویسید و scheduler موجود در xv۶ را به گونه ای تغییر دهید که به یک پروسه که محدودیت خود را رد کرده اجازه schedule شدن ندهد.

۱.۲.۱ syscall های مورد نیاز

برای این قسمت باید syscall زیر به xv۶ اضافه شوند:

```
int set_limit(int limit);
```

نباید امضای تابع [1] باقی syscall ها تغییر کند ولی تغییر بدنه آن ها مشکلی ندارد. (دقت کنید در قسمت بعد این syscall قرار است تغییر کند.)

۲.۲.۱ برنامه های سمت کاربر

باید برنامه ای بنویسید که سمت کاربر اجرا شود و با استفاده از آن تاثیر محدود کردن CPU را نشان دهید. برای اینکه تاثیر مشهود باشد یک کار CPU intensive، برای مثال یک حلقه for طولانی در نظر بگیرید و آن را چندبار با لیمیت های مختلف اجرا کنید.

۳.۱ Ram

برای این قسمت باید کد XV۶ را به گونه ای تغییر دهید که اگر میزان رم درخواستی یک پروسه از حدی که تعیین شده بیشتر شد، حافظه به آن اختصاص نیابد. برای سادگی فقط حافظه heap را در نظر بگیرید و کافیست کد تابع malloc موجود در فایل umalloc.c را به گونه ای تغییر دهید که میزان حافظه ای که کنون با استفاده از malloc تخصیص یافته هیچوقت از میزان محدودیت تعیین شده بیشتر نشود. همچنین باید تابع free را نیز بگونه ای تغییر دهید تا حافظه تخصیص داده شده ای که کاربر آزاد کرده، از میزان مصرف آن کم شود و در محدودیت لحاظ نشود.

۱.۳.۱ syscall های مورد نیاز

سمت کرنل باید syscall قبلی به طوری تغییر کند که امکان تعیین حداکثر RAM باشد، همچنین یک sycal جدید هم باید پیاده شود تا توابع malloc و free که در User space در حال اجرا هستند، بتوانند به کرنل اطلاع دهند که میزان مصرف رم کم و یا زیاد شده و اگر لیمیت را رد کرده، خروجی syscall باید -1 باشد.

[1] Function Signature

```
int set_limit(int cpu_quota, int memory_quota);
int increase_memory_usage(int amount);
```

(برای آزاد کردن ram یا میتوانید جوری پیاده سازی کنید که تابع increase_memory_usage ورودی منفی بگیرد و یا یک syscall جدید پیاده سازی کنید.)

۲.۳.۱ برنامه های سمت کاربر

باید یک برنامه بنویسید که به تعداد معین شده (مثلا ۱۰) بار حافظه malloc کند و در آخر برنامه تعداد allocation های موفق و ناموفق را بنویسد. سعی کنید که محدودیت های مختلف و تاثیر آن ها را تست کنید.

۲ آشنایی با xv6

چالش اصلی شما در این پروژه، آشنایی با XV6 ، چگونگی کارکرد بخش های مختلف و هدف فایل های مختلف آن است. به همین منظور، میتوانید از لینک های زیر استفاده کنید.

۱.۲ منابع

- اضافه کردن یک اپ سمت کاربر در XV6 لینک
- اضافه کردن یک syscall به XV6 (یه مسئله که تو این لینک گفته نشده، اینه که امضای تابع رو باید تو فایل defs.h هم اضافه کنید.)
- بخش های مرتبط این کتاب میتونه مفید باشه (مباحثش زیاده ولی قسمت های مرتبط رو بخونید) لینک
- این هم فورک ریپو XV6 هست که خودم یه مقدار تغییر دادم که یک سری مشکلاتی که موقع compile ممکن هست بخورید رو نداشته باشید. میتونید جهت راحتی از این ریپو به جای ریپو اصلی XV6 استفاده کنید. لینک

۲.۲ نکات

- هدف این پروژه این نیست که یک سیستم عامل بهینه و کارا بزنید، بلکه این است که یک پیاده سازی عملی از مفاهیمی که در کلاس یاد گرفته اید را ببینید. بنابراین بر روی بهینه بودن کد وسواس به خرج ندهید و هر کدی که نیازمندی های پروژه را برآورده کند کافی است.
- محدودیت ها باید به صورت Hard limit باشند. به این معنا که اگر یک پروسه محدودیت خود را رد کرد، حتی اگر CPU کار دیگری نداشته باشد نباید در آن ثانیه آن پروسه را رد کند. دررابطه با RAM صرفا کافیست که تابع malloc در صورتی که نتواند حافظه را اختصاص دهد خروجی ۰ بدهد.
- در آخرین ورژن های لینوکس، ممکن است نتوانید XV6 را کامپایل کنید، اگر مشکل داشتید یا از VM استفاده کنید یا اگر از ریپو فورک شده استفاده میکنید و داکر را نصب دارید میتوانید با استفاده از دستور زیر پروژه را کامپایل کنید:

```
make docker
```

۳ ارزیابی

ارزیابی فاز اول پروژه بر اساس پیاده سازی انجام داده شده و تسلط به مفاهیم و ساختار xv۶ است.