



سیستم‌های عامل (۴۰۴۲۴) - گروه ۲  
نیم‌سال دوم سال تحصیلی ۱۴۰۱-۱۴۰۲  
استاد درس: دکتر رسول جلیلی

تمرین دوم

مهلت تحویل: ۲۳:۵۹ روز جمعه ۱۴۰۲/۰۲/۱۵

نکات و قواعد

۱. محل بارگذاری تمرین تا سه روز پس از مهلت تحویل باز خواهد بود. در طول ترم، برای تکالیف عملی و تئوری در مجموع می‌توانید از ۱۲ روز تاخیر مجاز به صورت ساعتی استفاده کنید و پس از آن به ازای هر روز ۲۵ درصد جریمه بر روی نمره‌ی کسب شده اعمال خواهد شد.
۲. لطفا حتماً **آداب‌نامه‌ی انجام تمرین‌های درسی** را رعایت نمایید. در صورت استفاده از هر مرجعی و یا همفکری برای پاسخ به سوالات، مرجع مربوطه و یا نام همفکران را در پاسخ خود ذکر کنید.
۳. در صورتی که پاسخ سوالات را به صورت دست‌نویس آماده کرده‌اید، لطفاً تصاویر واضحی از پاسخ‌های خود ارسال کنید. در صورت ناخوانا بودن پاسخ ارسالی، نمره‌ای به پاسخ ارسال شده تعلق نمی‌گیرد.
۴. فایل مربوط به پاسخ خود را به فرمت `OS_HW۲_StdNum_FirstName_LastName` نامگذاری کرده و ارسال نمایید.
۵. سوالات خود را می‌توانید به این **آدرس** ارسال کنید.

سوال ۱ (۱۶ نمره)

- درست یا نادرست بودن هر کدام از عبارت‌های زیر را با ذکر دلیل مشخص کنید.
- (الف) ریسمان‌های سطح کاربر و سطح هسته از طریق فراخوانی سیستمی به هم سرویس می‌دهند.
- (ب) همه ریسمان‌های یک پرده، به فضای آدرسی هم دسترسی دارند و می‌توانند در فضای آدرسی هم بنویسند.
- (ج) ریسمان‌های سطح هسته به فضای آدرس ریسمان‌های سطح کاربر دسترسی دارند ولی ریسمان‌های سطح کاربر به فضای آدرس ریسمان‌های سطح هسته دسترسی ندارند (همه ریسمان‌ها را ریسمان‌های یک پرده در نظر بگیرید).
- (د) زمان‌بندی ریسمان‌های سطح کاربر سریع‌تر از زمان‌بندی ریسمان‌های سطح هسته است.

سوال ۲ (۱۵ نمره)

فرض کنید در هر لحظه فقط یک ریشه زباله روبی<sup>۲</sup> در حال اجرا است و برنامه تک ریشه است. آیا ترتیب قفل کردن در کد زیر، انحصار متقابل را برای نشان‌گر سراسری heap تضمین می‌کند یا خیر؟ دلیل خود را توضیح دهید.

```
int lock;
void *heap;

void *malloc(unsigned nbytes) {
    /* ... */
    lock = 1;
    heap = (char *)heap + nbytes;
    lock = 0;
    /* ... */
}
```

```
void initiate_gc(void) {
    if(lock)
        return;
    /* find free memory */
    heap = new_heap; /* reset heap */
}
```

## سوال ۳ (۱۸ نمره)

با توجه به جدول زیر، به سوالات پاسخ دهید (واحدهای زمانی جدول بر حسب میلی ثانیه هستند).

اولویت	مدت زمان انفجار <sup>۳</sup>	پروژه
۲	۲	$P_1$
۱	۵	$P_2$
۴	۱	$P_3$
۳	۷	$P_4$

(الف) نمودار gantt را برای الگوریتم‌های زمان‌بندی SJF و RR با عدد کوانتوم ۲ میلی ثانیه رسم کنید (در الگوریتم RR فرض کنید پروژه‌ها در ابتدا به ترتیب شماره از ابتدا تا انتهای Ready Queue قرار دارند).

(ب) فرض کنید زمان ورود پروژه‌ها به ترتیب ۰ و ۱ و ۲ و ۳ میلی ثانیه باشد. در این صورت نمودار gantt را برای الگوریتم Priority Scheduling که از نوع preemptive هست، رسم کنید.

(ج) برای هر سه الگوریتم، قسمت‌های الف و ب، میانگین زمان بازگشت<sup>۴</sup> و میانگین زمان انتظار<sup>۵</sup> را محاسبه کنید.

## سوال ۴ (۱۲ نمره)

در یک سیستم زمان‌بندی MLFQ دو صف RR با برش زمانی ۳ و بی‌نهایت وجود دارد. میانگین زمان بازگشت برای پروژه‌های زیر را محاسبه کنید؟

	Arrival Time	CPU Burst 1	I/O Burst 1	CPU Burst 2
$P_1$	۰	۳	۶	۲
$P_2$	۳	۷	۵	۱
$P_3$	۴	۲	۴	۴
$P_4$	۱۱	۵	۳	۱

## سوال ۵ (۱۲ نمره)

قطعه کد زیر را در نظر بگیرید. فرض کنید دو ریسره در حال اجرای تابع A به صورت هم‌روند هستند. آیا ممکن است متغیر X مقداری بیش‌تر از ۱۰ داشته باشد؟ توضیح دهید.

```
int x = 0;
void A()
{
    int r;
    while(x != 10)
    {
        r = x;
        x = r+1;
    }
}
```

## سوال ۶ (۱۲ نمره)

درستی یا نادرستی هر یک از عبارات زیر را مشخص کنید و در مورد دلیل نادرستی گزینه‌های نادرست توضیح دهید.

(الف) در صورتی که زمان اجرای همه پردازها برابر باشد و ترتیب ورود پردازها یکسان باشند، دو زمان‌بند Round-Robin و FCFS مانند یکدیگر عمل می‌نمایند.

(ب) با افزایش سطح چندبرنامگی<sup>۶</sup> مبتنی بر اشتراک زمانی<sup>۷</sup> کارایی CPU به صورت غیرخطی افزایش و سپس به صورت غیرخطی کاهش می‌یابد.

(ج) بر مبنای مقایسه متوسط زمان پاسخ، الگوریتم FIFO در بعضی مواقع بهتر از SJF عمل می‌کند.

(د) با توجه به جدول زیر، متوسط زمان انتظار پردازها برای الگوریتم Preemptive Shortest Remaining Job First برابر با ۳ واحد زمانی می‌باشد.

زمان ورود	زمان محاسبات	پردازه
۳	۵	$P_1$
۱	۸	$P_2$
۲	۶	$P_3$

## سوال ۷ (۱۵ نمره)

فرض کنید دو پردازه  $P_1$  و  $P_2$  در زمان ۰ آماده زمان‌بندی در یک سیستم توسط الگوریتم زمان‌بندی Round-Robin با ذره زمانی برابر با ۲ باشند. با فرض این که هر کدام از پردازها دارای ریسمان‌هایی به شرح زیر باشند:

پردازه  $P_1$  با  $T_{11}$  با زمان اجرای ۱,۵ و  $T_{12}$  با زمان اجرای ۱,۵  
پردازه  $P_2$  با  $T_{21}$  با زمان اجرای ۲,۵ و  $T_{22}$  با زمان اجرای ۲  
منظور از  $T_{ij}$ ، ریسمان  $i$ ام پردازهی  $j$ ام می‌باشد.

اگر الگوریتم زمان‌بندی ریسمان‌های درون هر پردازه LCFS: Last-Come First-Served باشد و ریسمان اول هر پردازه در لحظه آغاز اجرای آن و ریسمان دوم هر پردازه پس از یک واحد زمانی از لحظه آغاز به کار آن پردازه به سیستم وارد شوند، متوسط زمان بازگشت ریسمان‌های پردازه‌های  $P_1$  و  $P_2$  را محاسبه کنید.

موفق باشید