

آزمون میانی درس سیستم‌های عامل

مجموع نمره‌ها: ۱۰۰

زمان آزمون: ۸۰ دقیقه

- ۱ گاهی به سیستم عامل، ماشین مجازی (Virtual Machine) برای پردازش‌ها (Processes) هم گفته می‌شود. (۳)
توضیح دهید چه شباهتی بین سیستم عامل و ماشین مجازی وجود دارد؟
- ۲ «Core dump» چیست و چگونه به حل مشکلات موجود در برنامه‌ها کمک می‌کند؟ (۳)
- ۳ معماری میکروکرنل (Microkernel) را از سه جنبه ارزیابی کنید: الف) سختی گسترش سیستم عامل (برای مثال برای اضافه کردن راه‌اندازهای جدید)؛ ب) مقاومت در برابر خطاهای نرم‌افزاری در سیستم عامل (برای مثال دسترسی بد به حافظه در راه‌اندازها)؛ ج) سربارهای (Overheads) داخلی سیستم عامل (برای مثال برای خواندن از یک فایل). از همین سه جنبه، معماری‌هایی که از ماژول‌ها (Loadable Kernel Modules) استفاده می‌کنند را با میکروکرنل‌ها مقایسه نمایید. (۹)
- ۴ توضیح دهید وقفه‌های سخت‌افزاری به چه شکلی در زمانبندی در سیستم‌های عامل اشتراک زمانی (Time sharing) استفاده می‌شوند. (۳)
- ۵ در شبه کد زیر فراخوانی سیستمی «fork()» پردازش جدیدی ایجاد می‌کند و مقدار صفر را در پردازش فرزند و شماره‌ی پردازش فرزند (مقداری بزرگتر از صفر) را در پردازش پدر بر می‌گرداند. درخت پردازش‌ها را نمایش دهید (به هر پردازش یک رأس تخصیص دهید و پدر را با یال‌های جهت‌داری به فرزندانش متصل کنید). (۱۰)

```
int main(void)
{
    if (fork()) {
        fork();
        fork();
    } else {
        fork();
    }
    return 0;
}
```

- ۶ دو حالتی که پردازش‌های استفاده‌کننده از لوله (Pipe) در حالت «Blocking» قرار می‌گیرند را توضیح دهید. (۶)
- ۷ استفاده از «Thread pool» (تعدادی ریسمان آماده‌باش، برای اجرای وظیفه‌های درخواست شده) را از سه جنبه نسبت به استفاده‌ی مستقیم از ریسمان‌ها ارزیابی کنید (برای ادعاهای خود دلیل بیاورید): الف) مدیریت منابع سیستم عامل؛ ب) سربار ساختن ریسمان‌ها؛ ج) سختی مدیریت ریسمان‌ها. (۹)

۸ یک سرور فایل (File server) را در نظر بگیرید: برای هر کاربر، یک فایل (که قبلاً در حافظه‌ی اصلی قرار گرفته است) اول رمزنگاری می‌شود و دوم از راه شبکه به کاربر انتقال می‌یابد. استفاده از ریسمان‌ها چگونه کارایی این سرور را (با تعداد زیادی کاربر) بهبود می‌دهد؟ دو جنبه بیان نمایید.

۹ تابع `found()` توسط چند ریسمان صدا زده می‌شود و `best_cost` و `best_node` متغیر سراسری (Global) هستند. با مثالی توضیح دهید چرا مشکل «Race condition» پیش می‌آید و منجر به چه نتیجه‌ی غیر منتظره‌ای می‌شود. مشکل را با استفاده از قفل `Mutex` یا `Semaphore` حل نمایید.

```
void found(int node, int cost)
{
    if (cost < best_cost) {
        best_cost = cost;
        best_node = node;
    }
}
```

۱۰ راه حلی برای مسئله‌ی خواننده‌ها-نویسنده‌ها (Readers-Writers) ارائه دهید که در آن قحطی (Starvation) رخ ندهد (نه برای خواننده‌ها و نه برای نویسنده‌ها). در بیان راه حل خود باید حالت‌های ممکن برای انتظار خواننده‌ها و نویسنده‌ها را در نظر بگیرید و عملی که باید در هر یک از این حالت‌ها انجام شود را بیان کنید. سپس، راه حل خود را با استفاده از مانیتور پیاده‌سازی نمایید؛ مانیتور شما باید چهار تابع داشته باشد: `read_lock()` که قبل از خواندن از منبع، `read_unlock()` که پس از خواندن از منبع، `write_lock()` که قبل از نوشتن به منبع و `write_unlock()` که پس از نوشتن به منبع فراخوانی می‌گردند.

۱۱ در جدول زیر زمان اضافه شدن تعدادی پردازش به صف آماده‌باش (Ready queue) سیستم عامل نشان داده شده است. نمودار Gantt (نمایش وضعیت اجرای پردازش‌ها در طول زمان) را برای زمانبندی‌های FCFS (First-Come, First-Served)، SJF (Shortest Job First) و Priority (مقدار کوچک‌تر، اولویت بیشتر)، RR (Round-Robin) با برش‌های زمانی (Timeslice) ۱۰ میلی‌ثانیه نشان دهید. مقدار میانگین زمان Waiting و زمان Turnaround را برای هر سه الگوریتم محاسبه نمایید. الگوریتم‌های SJF و Priority «Nonpreemptive» هستند.

نام پردازش	زمان پردازش (میلی ثانیه)	زمان رسیدن (میلی ثانیه)	اولویت
A	۱۵	۰	۱
B	۵	۵	۳
C	۱۰	۱۰	۱
D	۵	۱۰	۲