

امتحان کلاسی اول درس سیستم عامل

۱ تابع `found()` مقدار بزرگترین عددی که به آن داده می‌شود را در متغیر `best` قرار می‌دهد. در صورتی که مقدار اولیه‌ی متغیر `best` برابر با صفر باشد و ریسمان اول این تابع را با پارامتر ۲۰ و ریسمان دوم آن را با پارامتر ۳۰ صدا بزند، آیا امکان رخداد وضعیت رقابتی وجود دارد؟ در این صورت توضیح دهید چگونه و مقدار متغیر `best` چگونه تغییر می‌کند.

```
int best = 0;
void f(int val)
{
    if (val > best)
        best = val;
}
```

۲ دو ریسمان را در نظر بگیرید و برای آنها چهار تابع زیر را با استفاده از سمافور یا مانیتور پیاده‌سازی کنید.

`enterA()` ریسمان اول قبل از ورود به ناحیه‌ی بحرانی خود این تابع را صدا می‌زند.

`leaveA()` ریسمان اول در هنگام خروج از ناحیه‌ی بحرانی این تابع را صدا می‌زند؛ در صورتی که ریسمان دوم

داخل ناحیه‌ی بحرانی خویش باشد، این تابع منتظر می‌ماند تا ریسمان دوم خارج شود.

`enterB()` ریسمان دوم قبل از ورود به ناحیه‌ی بحرانی خود این تابع را صدا می‌زند. در صورتی که ریسمان اول در

ناحیه‌ی بحرانی خود نباشد، این تابع منتظر می‌ماند تا ریسمان اول به ناحیه‌ی بحرانی خود وارد شود.

`leaveB()` ریسمان دوم در هنگام خروج از ناحیه‌ی بحرانی خود این تابع را صدا می‌زند.

امتحان کلاسی دوم درس سیستم عامل

توابع $f()$ و $g()$ توسط چند ریسمان به صورت همروند فراخوانی می‌شوند. آنها را با استفاده از سمافور یا مانیتور پیاده‌سازی کنید:

- . هر فراخوانی تابع $g()$ باید متوقف شود تا تابع $f()$ سه بار فراخوانی گردد (یا قبلاً فراخوانی شده باشد).
- . بعد از هر پانزده بار فراخوانی $f()$ ، این تابع باید منتظر باشد که تابع $g()$ پنج بار صدا زده شده باشد.