



سیستم عامل بهسید (بهترین سیستم عامل دنیا) که توسط شرکت خوخو طراحی شده است، قسمت بزرگی از بازار سیستم عامل‌های شهر نائل را تصاحب کرده است. برای مقابله با انحصار این شرکت، گروهی از دانشجویان علاقمند به سرکردگی ر.م. تصمیم گرفته‌اند با تولید سیستم عامل جدیدی به نام غوخو (غروب خوخو) بازار را از دست خوخو خارج کنند و با این کار مقدمه‌ی سقوط این شرکت را فراهم نمایند. به آنها در طراحی این سیستم عامل جاه طلبانه کمک کنید.

- ۱- طراحان غوخو در تصمیم ساختار این سیستم عامل دچار تردید شده‌اند. هدف غوخو، سیستم عاملی است که در مقابل خطاهای برنامه‌نویسی در راه اندازه‌ها تا حد امکان مقاوم باشد. از بین ساختارهای سیستم عامل، «با دلیل» بیان کنید که کدام نسبت به این نوع خطاها مقاوم‌تر است.
- ۲- در سیستم عامل غوخو قرار است از روش مبتنی بر درگاه (Port-based I/O) برای ورودی و خروجی استفاده شود. آیا روش دیگری برای انتقال اطلاعات بین پردازنده و دستگاه ورودی و خروجی وجود دارد؟ برای اطلاع برنامه‌نویسان غوخو، هر دو روش را توضیح دهید.
- ۳- در نمونه‌ی اولیه‌ی غوخو مشکلی وجود دارد که موجب سردرگمی برنامه‌نویسان آن به خصوص ر.م. شده است. آیا امکان دارد هم سر نوشتن و هم سر خواندن لوله (Pipe) همزمان در استفاده از آن منتظر (Block) شوند؟ اگر بله در چه هنگام و اگر خیر چرا؟
- ۴- در شبه کد زیر با فراخوانی تابع `fork()` سیستم عامل یک پردازش جدید از روی پردازش فراخوانی کننده ایجاد می‌کند؛ در پردازش فراخوانی کننده (پدر) این تابع شناسه‌ی پردازش جدید (مقداری بزرگ‌تر از صفر) و در پردازش جدید (فرزند) مقدار صفر را بر می‌گرداند. درخت پردازش‌هایی که ایجاد می‌شوند را بکشید (به هر پردازش یک رأس اختصاص دهید و پردازش پدر را با یال‌های جهت‌دار به فرزندانش وصل کنید).

```
int main(void)
{
    int i = 0;
    if (fork() > 0)
        i++;
    if (fork() > 0)
        i++;
    if (i == 1)
        fork();
    return 0;
}
```

- ۵- بیست پردازش در سیستم عامل غوخو اجرا می‌شوند. با فرض اینکه زمان تعویض متن (Context switch) دو میلی‌ثانیه باشد، مقدار بیشینه‌ی برش زمانی (Time slice) در الگوریتم Round-Robin را به صورتی تعیین کنید که زمان پاسخگویی حداکثر دو بیست میلی‌ثانیه باشد.



- ۶- در شبه کد زیر، تابع admit توسط چند بند به صورت همروند فراخوانی می‌شود و متغیر cnt سراسری است. این تابع، باید دقیقاً سه بار مقدار یک و در سایر فراخوانی‌ها مقدار صفر را برگرداند. رخداد وضعیت رقابتی را با بیان دنباله‌ای از فراخوانی‌ها و مقدار متغیر cnt نشان دهید.

```

1  int cnt = 3;
2  int admit(void)
3  {
4      //
5      if (cnt > 0) {
6          cnt = cnt - 1;
7          //
8          return 1;
9      }
10     return 0;
11 }
```

- ۷- در پرسش قبل، برای حل مشکل وضعیت رقابتی، یک سمافور با نام sem و با مقدار اولیه‌ی یک تعریف، در خط چهارم عبارت wait(sem); و در خط هفتم عبارت signal(sem); قرار داده می‌شود. اگر مشکل حل می‌شود، بیان کنید چرا و اگر حل نمی‌شود، بیان کنید چه مشکلی رخ می‌دهد؟

- ۸- در غوخو، از یک الگوریتم زمانبندی سه صفت استفاده می‌شود. در صف اول از زمانبند Round Robin با برش زمانی پنج میلی‌ثانیه، در صف دوم از الگوریتم زمانبندی Shortest Job First، در صف سوم از الگوریتم First-Come First-Served و بین صف‌ها از زمانبند اولویت استفاده می‌شود. نمودار Gantt را برای زمانبندی پرده‌های جدول زیر بکشید (زمان‌ها به میلی‌ثانیه هستند).

پرده	صف	زمان ورود	زمان پردازش
A	۲	۰	۲۰
B	۳	۵	۱۰
C	۱	۵	۱۰
D	۲	۱۵	۵

- ۹- در سیستم عامل غوخو لازم است قفلی پیاده‌سازی شود که اجازه دهد در هر لحظه حداکثر سه بند بتوانند آن را قفل کنند (ناحیه‌ی قفل شده می‌تواند توسط سه بند به طور همزمان استفاده شود). برای این کار، دو تابع lock3() و unlock3() را به کمک سمافور یا مانیتور پیاده‌سازی نمایید؛ تابع اول قفل می‌کند و تابع دوم آن را باز می‌کند.