

## 第六题 (10 分)

1. 火锅洞洞主终于要请吃火锅啦! 由于防疫规定, 洞主决定分批邀请, Alice、Bob、……、Zach 这 26 位同学顺理成章地成为了第一批受到邀请的同学。他们围坐在一张圆桌旁, 按照首字母顺序排列 (即 Alice 的右边是 Bob, Bob 的右边是 Carol, …… , Zach 的右边是 Alice)。吃饭之前, 洞主想要验证他们的身份, 他提出了这样的要求:

- 1) 每位同学有一个编号, 即字母序号减 1, 也就是说, Alice 是 0 号, Bob 是 1 号, 以此类推。
- 2) 每位同学需要同时拿着旁边两位同学的手机去找洞主 (不必带上自己的手机), 按照洞主的要求在树洞里发信息以验证身份。
- 3) 可以同时有多位同学找洞主验证 (也就是说, Alice 需要拿着 Zach 和 Bob 的手机找洞主, 并且在同一时间, Bob 也可以拿着 Alice 和 Carol 的手机找洞主。但是, 如果 Alice 还没同时拿到 Zach 和 Bob 的手机, 就只能坐在座位上等待。)

所幸, 在座有多位字母君学过 ICS, 他们决定采用信号量解决问题。

1.1 Alice 给出了这样的代码, 其中 thread 函数的参数指明了字母君的编号:

```
1. #include "csapp.h"
2. #define N 26
3. #define UP(i) ((i + 1) % N)
4. #define DOWN(i) ((i - 1 + N) % N)
5. sem_t sem[N];
6.
7. void *thread(void *i){
8.     int num = (int)i;
9.     P(&sem[UP(num)]);
10.    P(&sem[DOWN(num)]);
11.
12.    /* verify identity */
13.
14.    V(&sem[UP(num)]);
15.    V(&sem[DOWN(num)]);
16. }
17. int main(){
18.     for (int i = 0; i < N; i++)
19.         Sem_init(&sem[i], 0, 1);
20.     pthread_t tid[N];
21.     for (int i = 0; i < N; i++)
22.         Pthread_create(&tid[i], NULL, thread, (void *)i);
23.     Pthread_exit(0);
24. }
```

Bob 一看, 皱起了眉头。请问这段代码可能会发生什么问题 (用一个专用名词表述即可)?

1.2 Carol 把 thread 函数换成了下面这样:

```
1. void *thread(void *i){
2.     int num = (int)i;
3.     for (int i = 0; i < N; i++)
4.         P(&sem[i]);
5.     /* verify identity */
6.     for (int i = 0; i < N; i++)
7.         V(&sem[i]);
8. }
```

Dave 指出, 这段代码固然正确, 但是效率太低了, 他认为只要把第 3、4 行的代码换成一个 P 操作, 把第 6、7 行的代码换成一个 V 操作, 就能实现同样的效果。请问 Dave 的方法是:

1.2.1 P 操作: P(&sem[\_\_\_\_\_]);

1.2.2 V 操作: V(&sem[\_\_\_\_\_]);

1.3 Eve 看了 Dave 的方案还是摇头, 这样执行效率太低了, 没能利用同时可以有多位同学找洞主验证的条件。借助“互斥锁加锁顺序规则”, 他给出了自己的代码。你能把代码补全吗?

```
1. void *thread(void *i)
2. {
3.     int num = (int)i;
4.     if (UP(num) < DOWN(num)){
5.         P(&sem[___A___]);
6.         P(&sem[___B___]);
7.     }
8.     else{
9.         P(&sem[___C___]);
10.        P(&sem[___D___]);
11.    }
12.    /* verify identity */
13.    V(&sem[UP(num)]);
14.    V(&sem[DOWN(num)]);
15. }
```

A:\_\_\_\_\_ B:\_\_\_\_\_ C:\_\_\_\_\_ D:\_\_\_\_\_

以上 4 空均填序号:

- ① UP(num)
- ② DOWN(num)
- ③ num

2. Hungary Alice 发现自己没吃上第一场的火锅, 一怒之下, 给上面的 26 位字母君出了一道难题: 用二元信号量实现信号量 (也就是值可以是任意非负整数的信号量)。精通 ICS 的你自然认为这是小儿科, 迅速补全了以下代码。

```

1. #include "csapp.h"
2. typedef struct{
3.     int value;
4.     sem_t mutex;
5.     sem_t zero;
6. } my_sem_t;
7. void my_sem_init(my_sem_t *sem, unsigned int value){
8.     sem->value = value;
9.     Sem_init(&sem->mutex, 0, __A__);
10.    Sem_init(&sem->zero, 0, __B__);
11. }
12. void my_P(my_sem_t *sem){
13.     P(&sem->mutex);
14.     sem->value--;
15.     if (sem->value __C__ 0)
16.     {
17.         __D__;
18.         __E__;
19.     }
20.     else
21.         V(&sem->mutex);
22. }
23. void my_V(my_sem_t *sem){
24.     P(&sem->mutex);
25.     sem->value++;
26.     if (sem->value __F__ 0)
27.     {
28.         V(&sem->mutex);
29.         V(&sem->zero);
30.     }
31.     else
32.         V(&sem->mutex);
33. }

```

A: \_\_\_\_\_ B: \_\_\_\_\_ C: \_\_\_\_\_  
 D: \_\_\_\_\_ E: \_\_\_\_\_ F: \_\_\_\_\_

D 和 E 请填序号:

- ① P(&sem->mutex)
- ② P(&sem->zero)
- ③ V(&sem->mutex)
- ④ V(&sem->zero)

得分

### 第七题（10 分）

解决第一类读者-写者问题的代码如下：

```
int readcnt;    /* Initially 0 */
sem_t mutex, w; /* Both initially 1 */
void reader(void)
{
(1)   while (1) {
(2)       P(&mutex);
(3)       readcnt++;
(4)       if (readcnt == 1) /* First in */
(5)           P(&w);
(6)       V(&mutex);
(7)       /* Reading happens here, need 7 time unit */
(8)       P(&mutex);
(9)       readcnt--;
(10)      if (readcnt == 0) /* Last out */
(11)          V(&w);
(12)      V(&mutex);
    }
}

void writer(void)
{
(13)  while (1) {
(14)      P(&w);
(15)      /* Writing here, need 8 time unit */
(16)      V(&w);
    }
}
```

假设有 5 个读者或写者到来，到达时刻和所需的时间如下：

线程	到达时刻	读写时间
R1	0	7
W1	1	8
R2	2	7
W2	4	8
R3	5	7

注意，为简单起见，只考虑读写时间，忽略所有其他时间（包括代码中其他语句的执行时间、线程切换、调度等等），亦假设没有更多的读者或写者或其他线程。

- (1) 在时刻 3 时，R1、W1 和 R2 所处的位置，请标出对应的代码行号。此刻，readcnt 和 w 的值分别是多少？R1: \_; W1: \_; R2: \_; readcnt: \_; w: \_。
- (2) 在时刻 6 时，W2 和 R3 所处的位置，请标出对应的行号。此刻，readcnt 和 w 的值分别是多少？W2: \_; R3: \_; readcnt: \_; w: \_。
- (3) 如果不使用 P(&mutex) 和 V(&mutex)，程序执行会不会出错？如果出错，会出什么错？