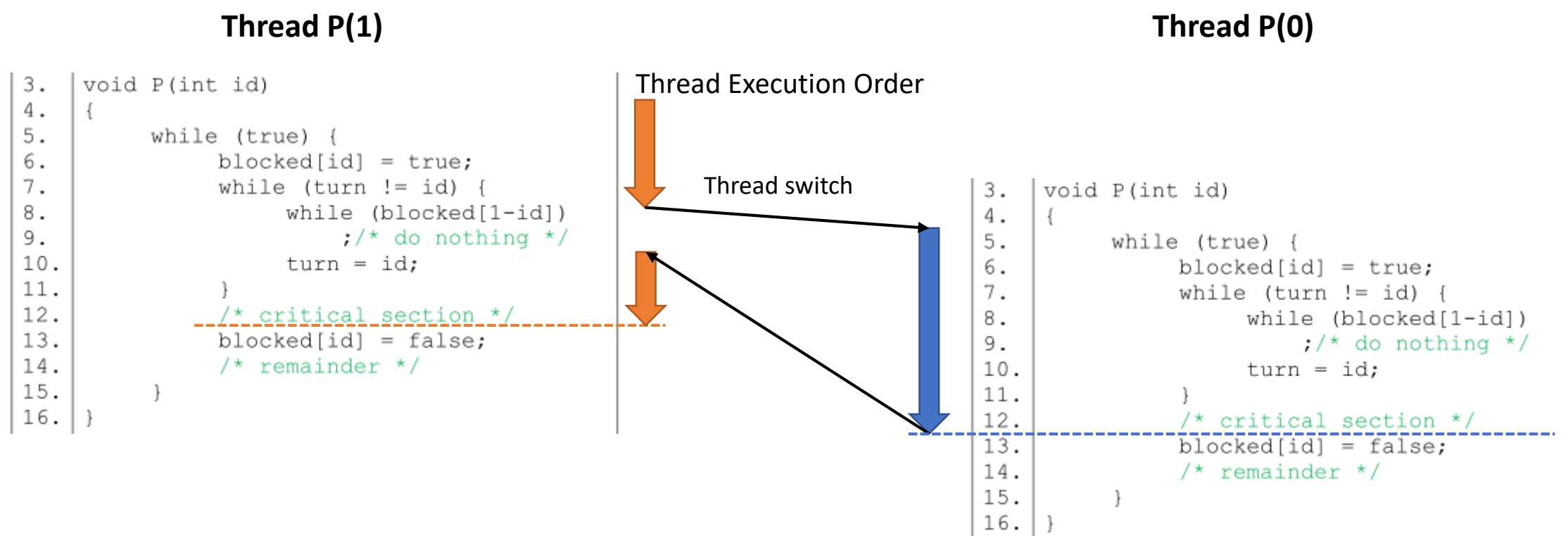


- 1.
- No. Consider the following case.
1. Initially `turn` equals 0.
2. P(1) sets `blocked[1]` to true (line 6), enters the 2nd while loop (line 7), and finds `blocked[0]` false, so skips the 3rd while loop (line 8). Next, P(1) switches out and P(0) switches in.
3. P(0) then sets `blocked[0]` to true (line 6), finds `turn = 0` (line 7), and enters its critical section (line 12). Next, P(0) switches out and P(1) switches in.
4. P(1) then assigns 1 to `turn` (line 10) and also enters its critical section (line 12).



2. a) wsem = -1, rsem = 0, x =1, y =1, z =1

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Writer

```
semWait(y)
    writecount++;
    if (writecount==1)
        semWait(rsem);
semSignal(y);

semWait(wsem);
WRITEUNIT( );
semSignal(wsem);

semWait(y);
    writecount--;
    If (writecount==0)
        semSignal(rsem);
semSignal(y);
```

2. b) $z = 0$, $rsem = -1$, others remain no change

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Writer

```
semWait(y)
    writecount++;
    if (writecount==1)
        semWait(rsem);
semSignal(y);

semWait(wsem);
WRITEUNIT( );
semSignal(wsem);

semWait(y);
    writecount--;
    If (writecount==0)
        semSignal(rsem);
semSignal(y);
```

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

2. c) $z = -1$, others remain the same

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Writer

```
semWait(y)
    writecount++;
    if (writecount==1)
        semWait(rsem);
semSignal(y);

semWait(wsem);
WRITEUNIT( );
semSignal(wsem);

semWait(y);
    writecount--;
    If (writecount==0)
        semSignal(rsem);
semSignal(y);
```

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

2. d) wsem=-2, others remain the same

2. e) the first writer

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Writer

```
semWait(y)
    writecount++;
    if (writecount==1)
        semWait(rsem);
semSignal(y);

semWait(wsem);
WRITEUNIT( );
semSignal(wsem);

semWait(y);
    writecount--;
    If (writecount==0)
        semSignal(rsem);
semSignal(y);
```

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Reader

```
semWait(z);
semWait (rsem);
semWait(x);
    readcount++;
    if(readcount==1)
        semWait(wsem);
semSignal(x);
semSignal (rsem);
semSignal(z);

READUNIT( );

semWait(x);
    readcount--;
    if (readcount==0)
        semSignal(wsem);
semSignal(x);
```

Writer

```
semWait(y)
    writecount++;
    if (writecount==1)
        semWait(rsem);
semSignal(y);

semWait(wsem);
WRITEUNIT( );
semSignal(wsem);

semWait(y);
    writecount--;
    If (writecount==0)
        semSignal(rsem);
semSignal(y);
```

Self-test

- 1. C
- 2. C
- 3. A
- 4. semWait and semSignal
- 5. A
- 6. B