(First)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);
```

(First) 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);
```

What will be the values of the semaphores when a writer wants to write while the first reader is reading?

rsem=0, wsem=-1, x=1, y=1, z=1

(First) 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);
```

(First) 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);
```

(Second) 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);
```

Continue with a), what will be the values of the semaphores when a second reader wants to read while the first reader is still reading?

z=0, rsem=-1, others remain the same (wsem=-1, x=1, y=1)

…

```
(First) 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);
```

```
(Second) 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);
```

```
(Third ) 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);
```

Continue with b), what will be the values of the semaphores when a third reader wants to read while the first reader is still reading?

z=-1, others remain no change (rsem=-1, x=1, wsem=-1, y=1)

…

```
(Second) 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);
```

```
(Third ) 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);
```

```
(Second) 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);
```

Continue with c), what will be the values of the semaphores when a second writer wants to write while the first reader is still reading?

wsem=-2, others remain no change (rsem=-1, x=1, y=1, and z=-1)

Which one will resume first when the first reader finishes reading, assuming all the semaphores are strong semaphores?

When the first reader finishes reading, calls semSignal(wsem), one of the two writers which are blocked on wsem will be unblocked. For strong semaphore following FIFO, the first writer will be unblocked.

```
(First) 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);
```

```
(First) 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);
```

```
(Second) 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);
```