

# 05\_19\_과제

## 2018320161\_컴퓨터학과\_송대선

int is\_wrt //버퍼에 접근하는 Writer 숫자

semaphore rea//Reader 간 동기화

초기값 : is\_wrt=0

### Writer 프로세스

```
is_wrt = 1
wait(wrt); // entry section
...
writing is performed
...
signal(wrt); //exit section
is_wrt = 0
```

### Reader 프로세스

```
wait(mutex);
if (is_wrt == 0)
    readcount++;
    if (readcount == 1)
        wait(wrt);

    signal(mutex);
...
reading is performed
...
wait(mutex);
    readcount--;
    if (readcount == 0)
        signal(wrt);
    signal(mutex);
else
    wait(rear)
```

버퍼에 접근하는 Writer 숫자 변수를 이용하여 writer이 버퍼에 접근하려고 하면 reader를 wait시킵니다.

웹검색을 통해 살펴본 자료들과는 기본적인 philosophy는 같으나, 구현의 방법이 크게 차이가 납니다.

찾아낸 하나의 예시를 아래에 첨부하여 봅니다.

```
class ReadWrite {
    ... // as before
    private protected
    int waitingW = 0; // no of waiting Writers.

    synchronized public void acquireRead() {
        while (writing || waitingW>0) {
            ... wait(); ...
        }
    }
}
```

```
    }  
    ++readers;  
}  
  
synchronized public void releaseRead() {  
    ...  
}  
  
synchronized public void acquireWrite() {  
    while (readers>0 || writing) {  
        ++waitingW;  
        ...  
        try{  
            wait();  
            ...  
            --waitingW;  
        }  
        writing = true;  
    }  
  
    synchronized public void releaseWrite() {... }  
}
```