Barrier Solution

```
1  n = the number of threads
2  count = 0
3  mutex = Semaphore(1)
4  barrier = Semaphore(0)
```

count: keeps track of how many threads have arrived

mutex: provides exclusive access to `count`

barrier: is locked (≤ 0) until all threads arrive

When `barrier.value<0`,

`barrier.value == Number of queueing processes`

```
1  specific_task();
2  mutex.wait();
3      count++;
4  mutex.signal();
5  if (count < n)
6      barrier.wait();
7  barrier.signal();
8  critical_point();
```

```
1  specific_task();
2  mutex.wait();
3      count++;
4  mutex.signal();
5  if (count == n)
6      barrier.signal();
7  barrier.wait();
8  critical_point();
```

Deadlock!

Only one thread can pass the barrier!