

# Concurrent Queue (Unbounded)

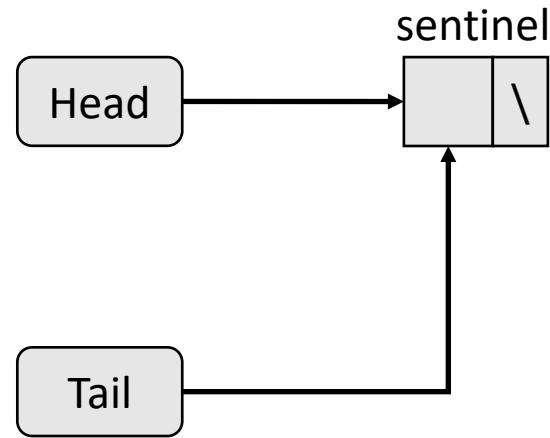
Concurrent Programming

# Introduction

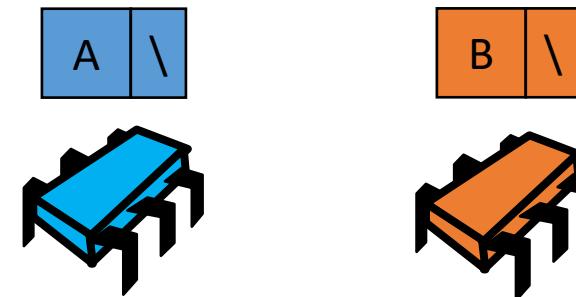
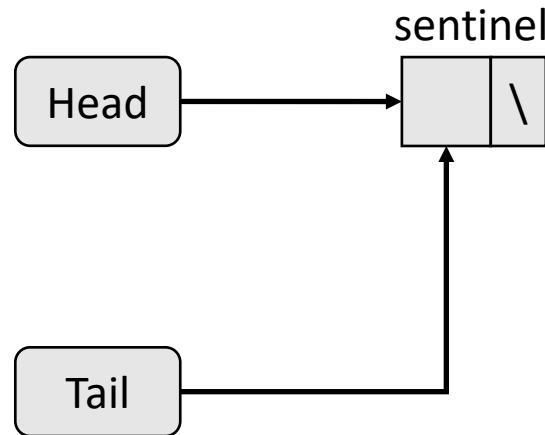
---

- Unbounded Lock-free Queue
- Evaluation
  - Queue with coarse-grained locking
  - Unbounded Lock-free Queue (covered in lecture note)
  - Bounded Lock-free Queue

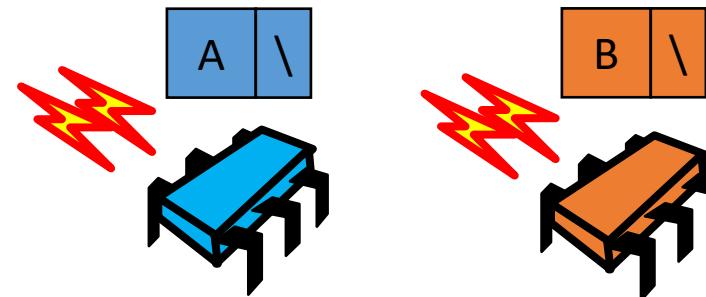
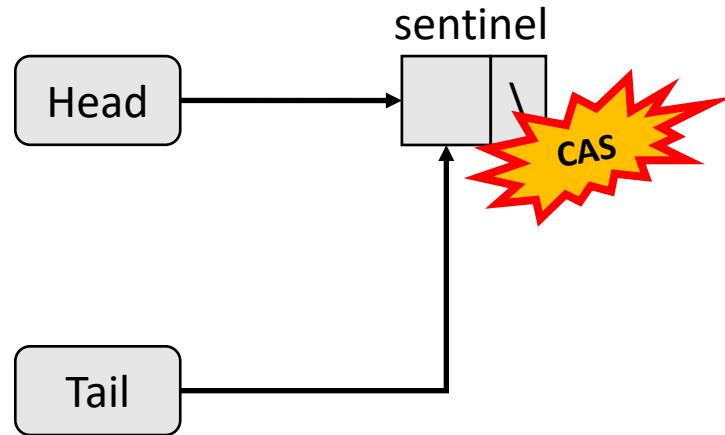
# Unbounded Lock-free Queue



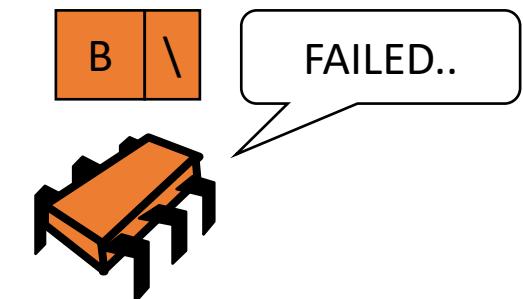
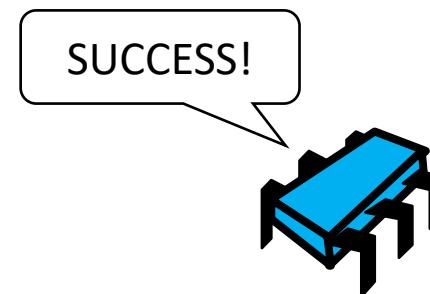
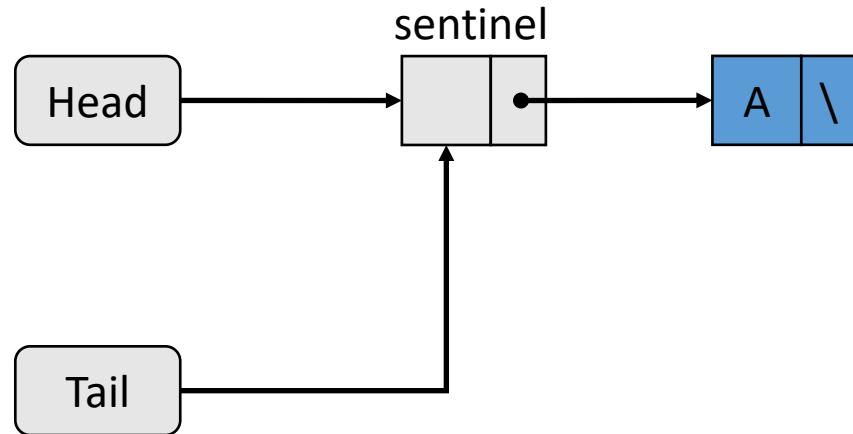
# Unbounded Lock-free Queue



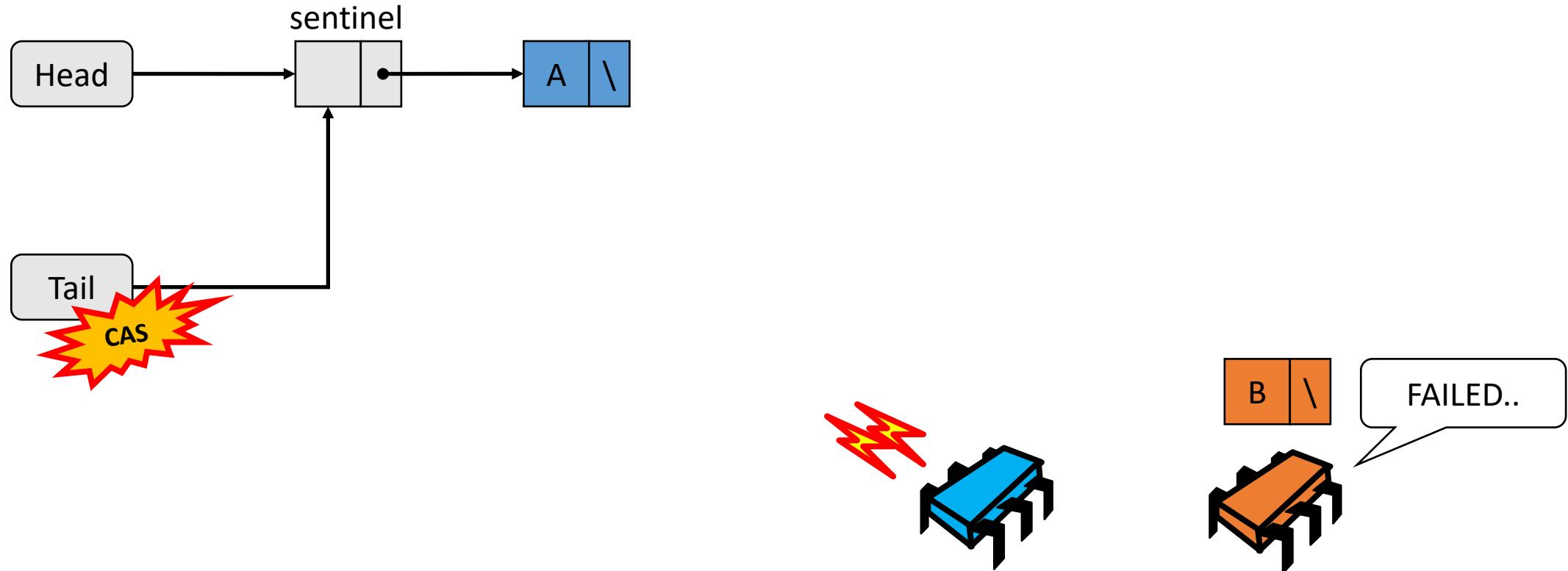
# Unbounded Lock-free Queue



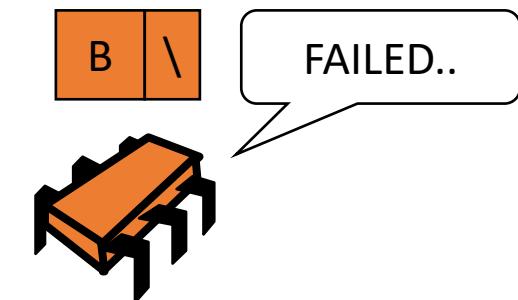
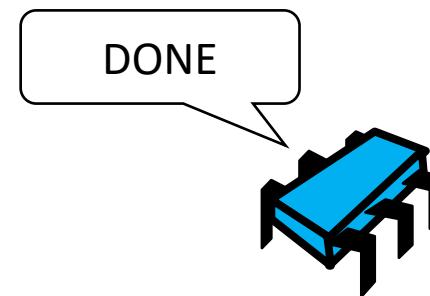
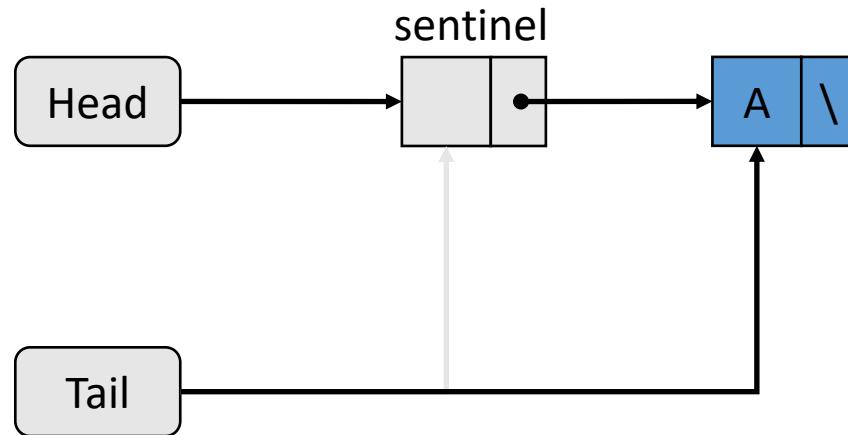
# Unbounded Lock-free Queue



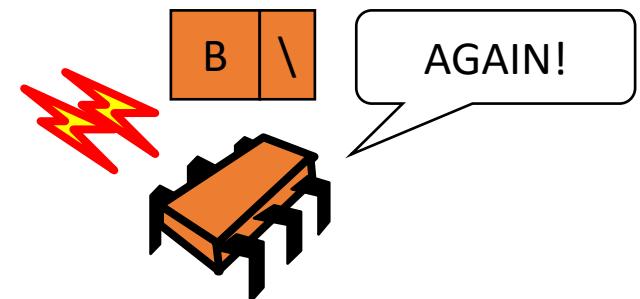
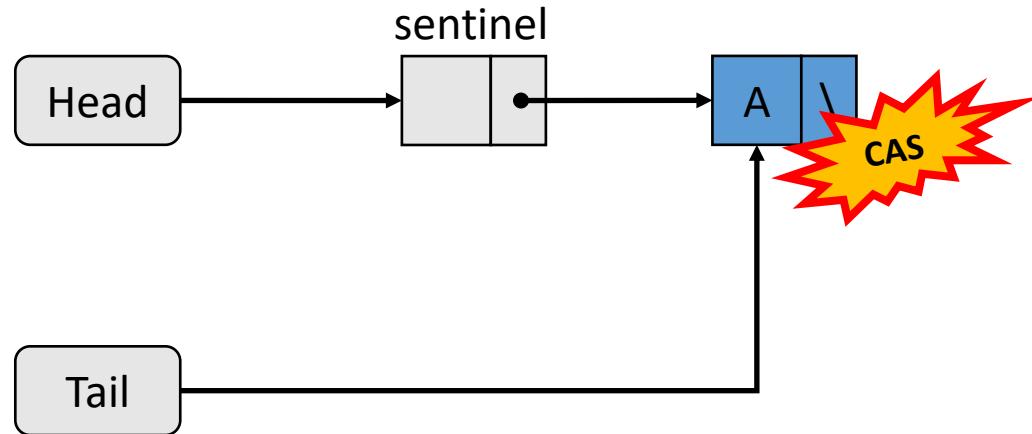
# Unbounded Lock-free Queue



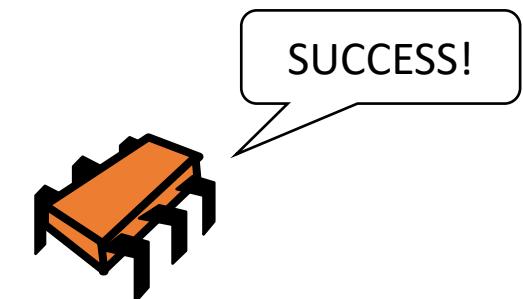
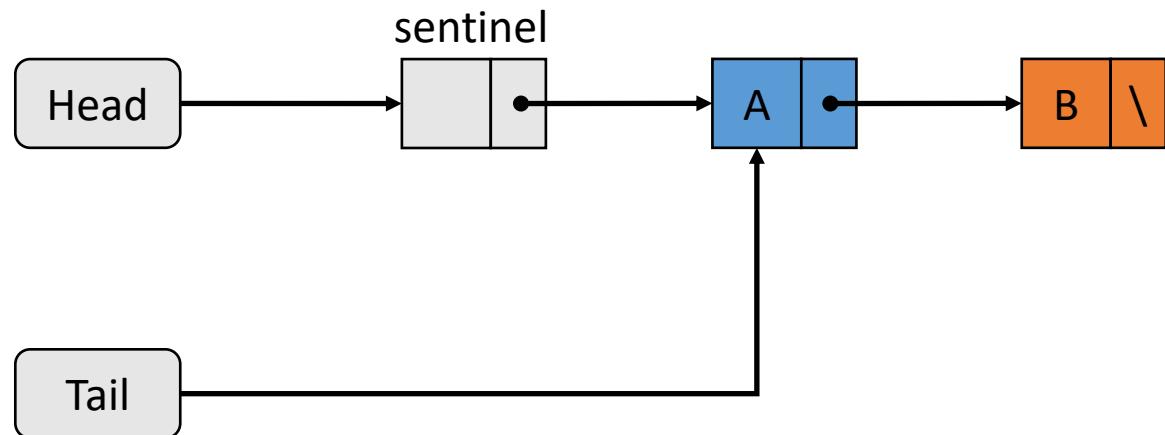
# Unbounded Lock-free Queue



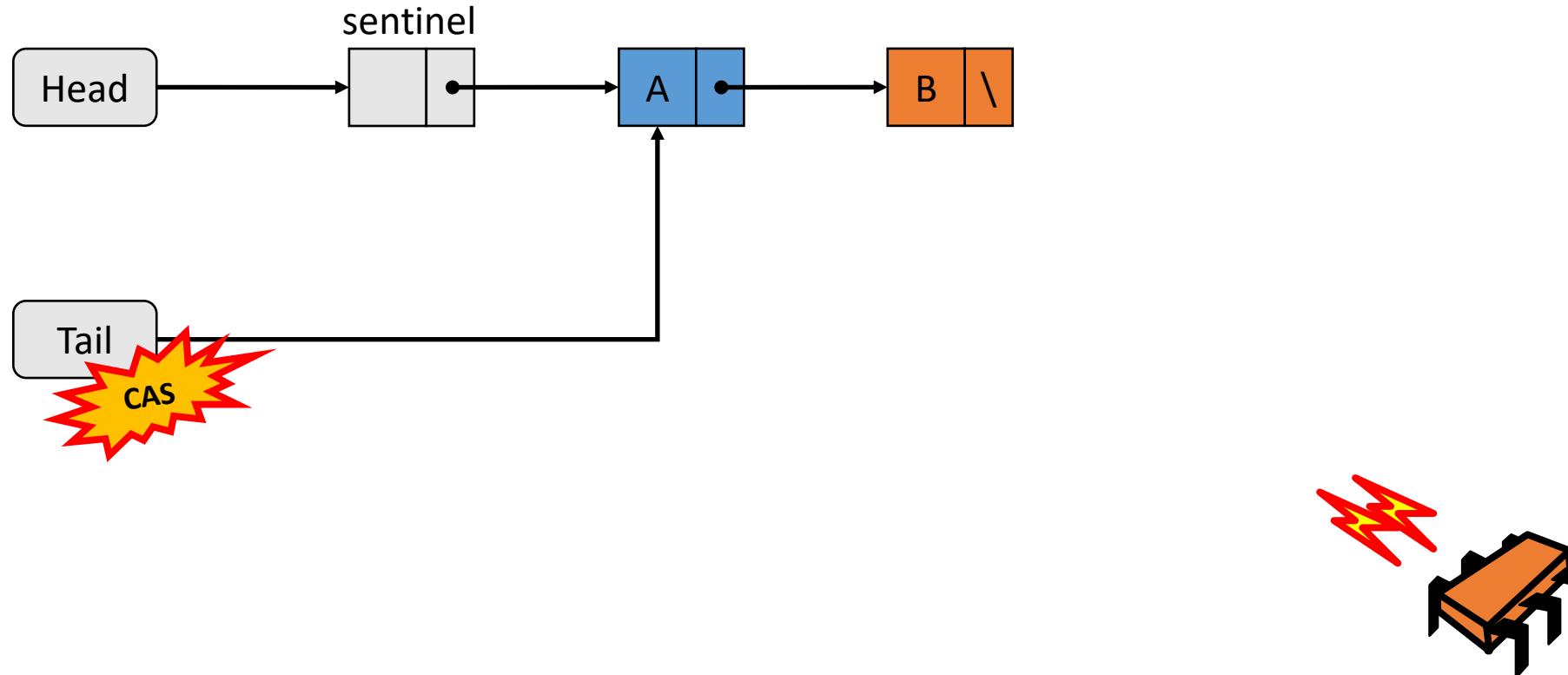
# Unbounded Lock-free Queue



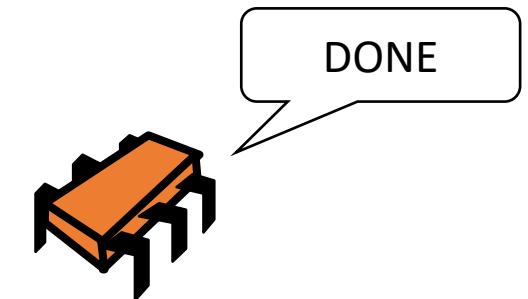
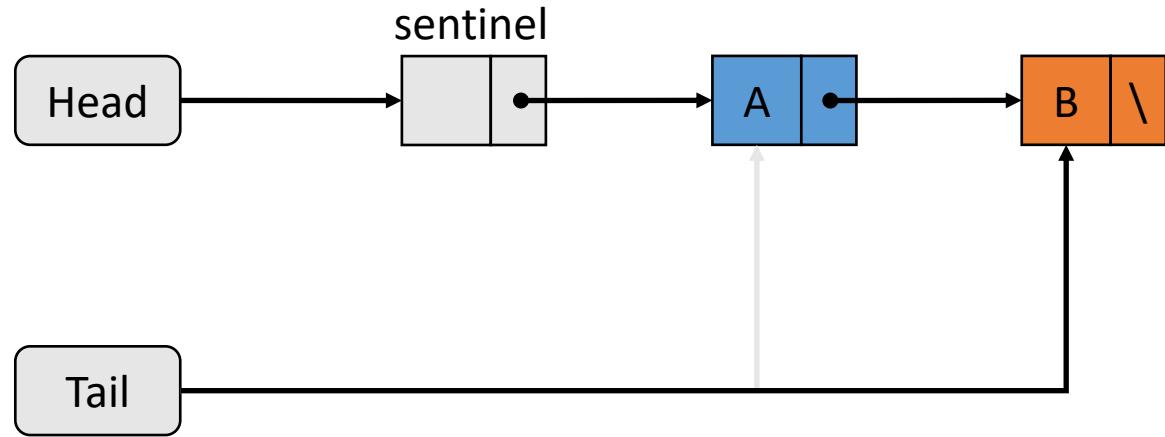
# Unbounded Lock-free Queue



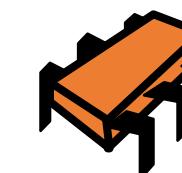
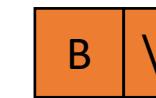
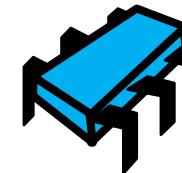
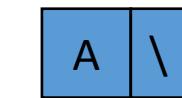
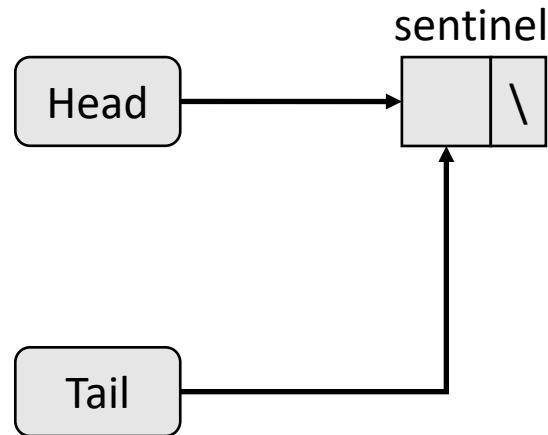
# Unbounded Lock-free Queue



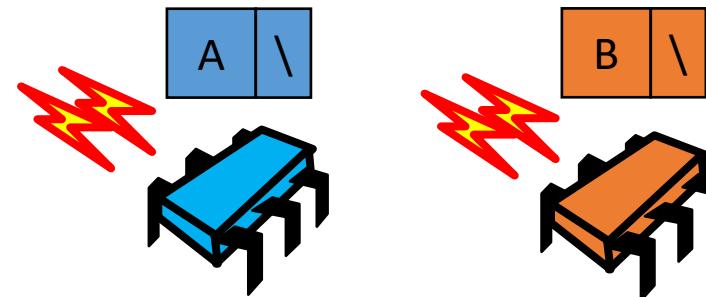
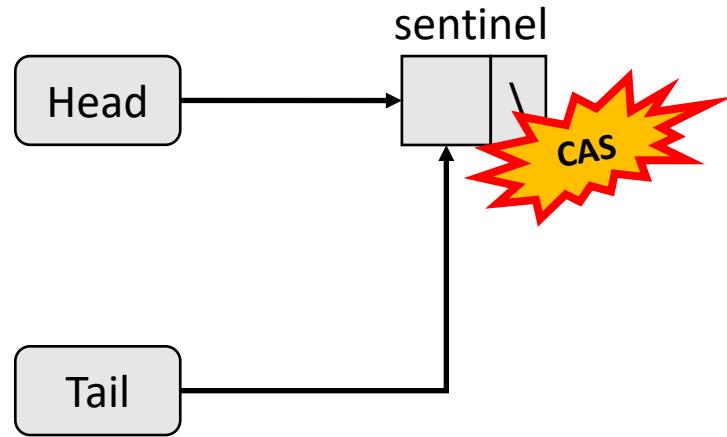
# Unbounded Lock-free Queue



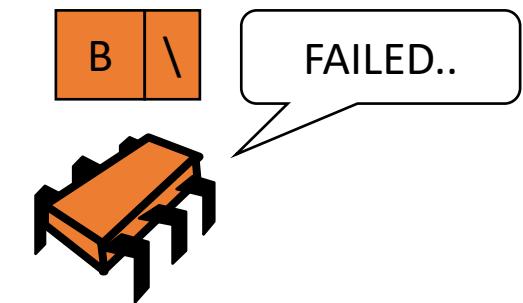
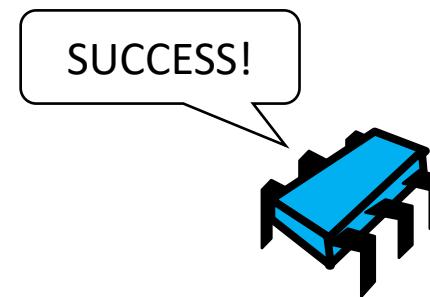
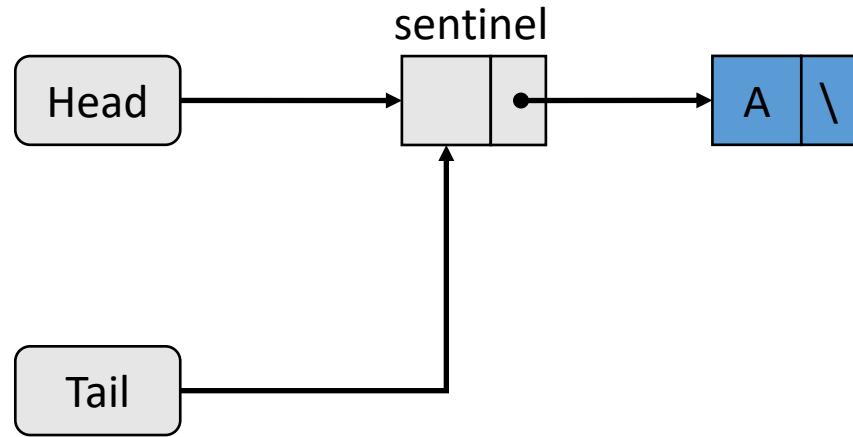
# Unbounded Lock-free Queue



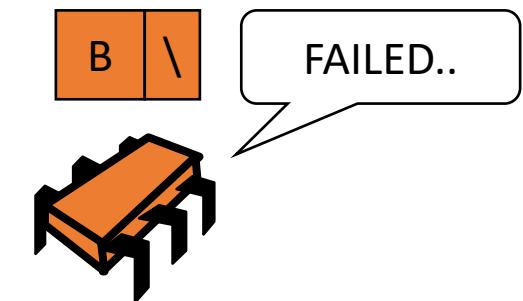
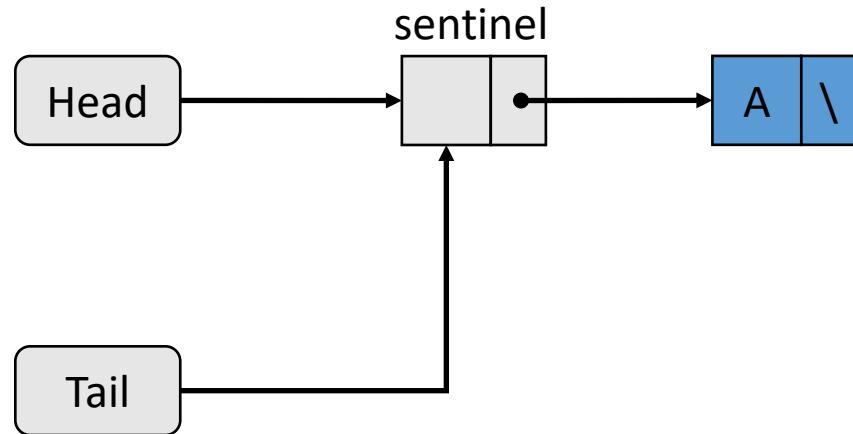
# Unbounded Lock-free Queue



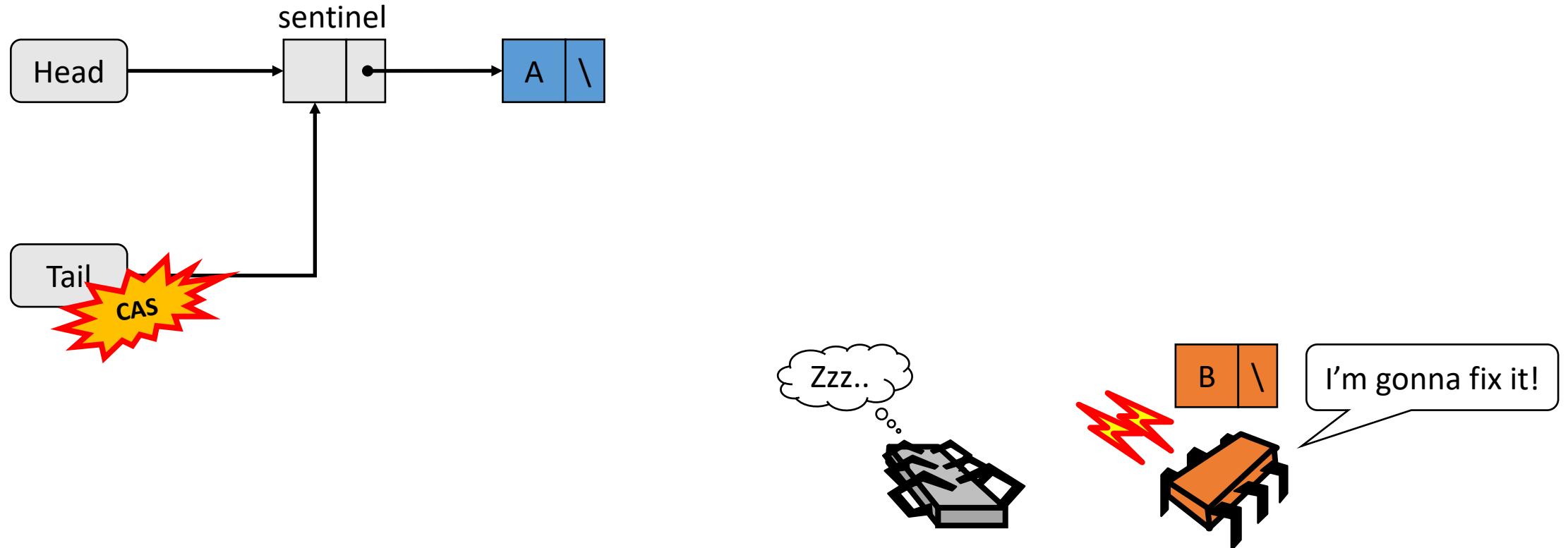
# Unbounded Lock-free Queue



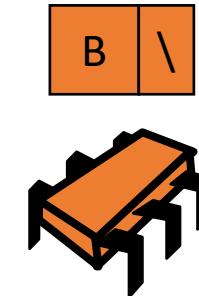
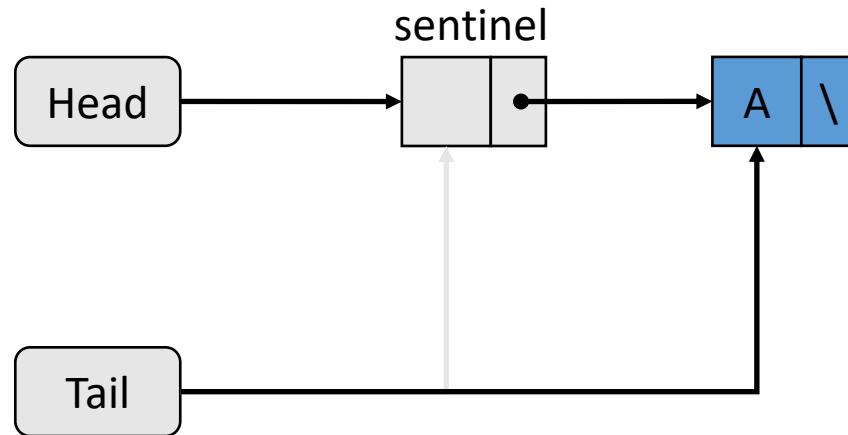
# Unbounded Lock-free Queue



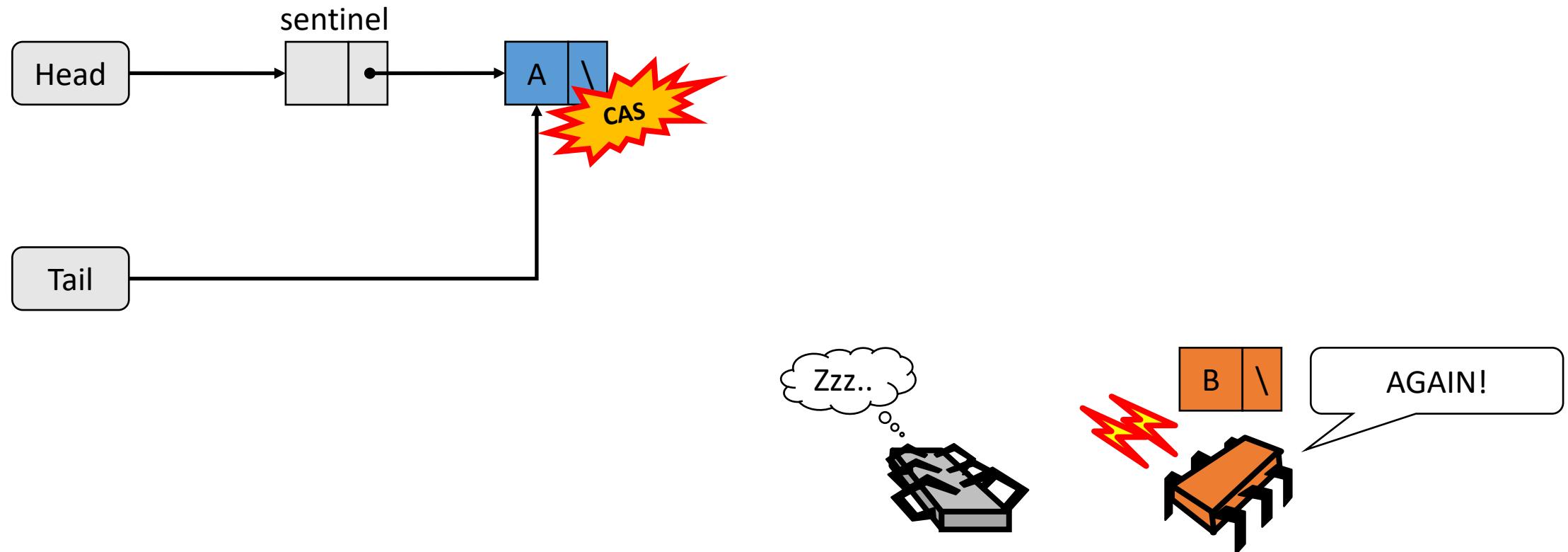
# Unbounded Lock-free Queue



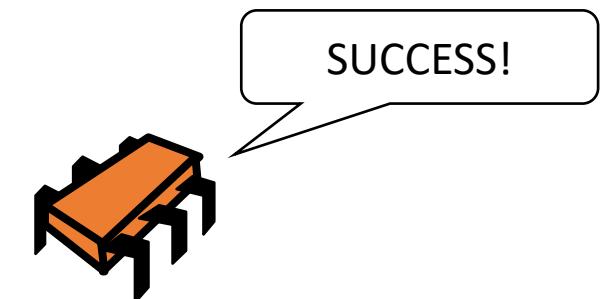
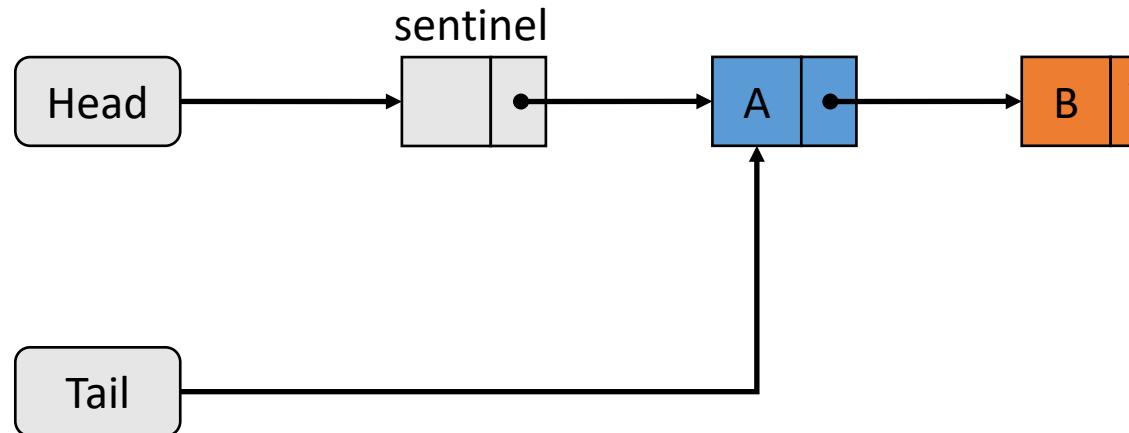
# Unbounded Lock-free Queue



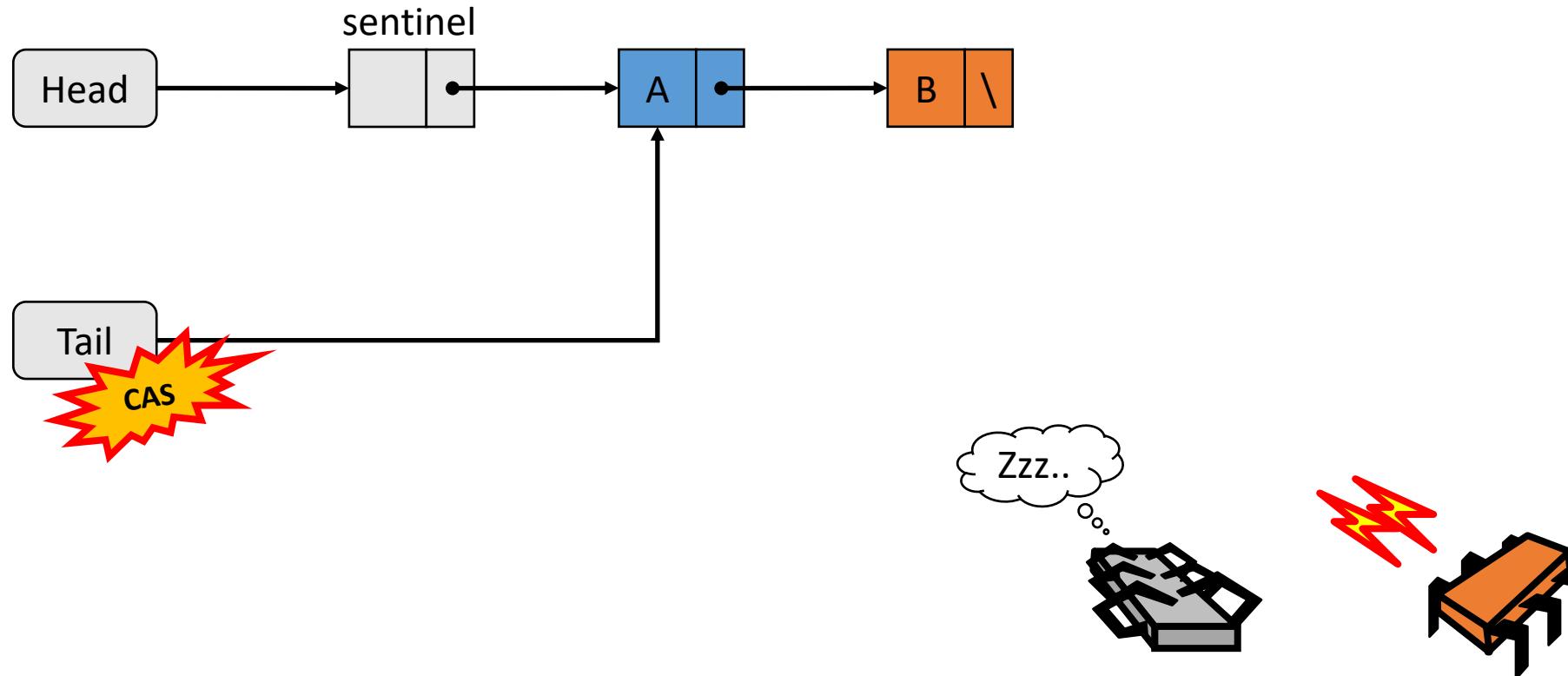
# Unbounded Lock-free Queue



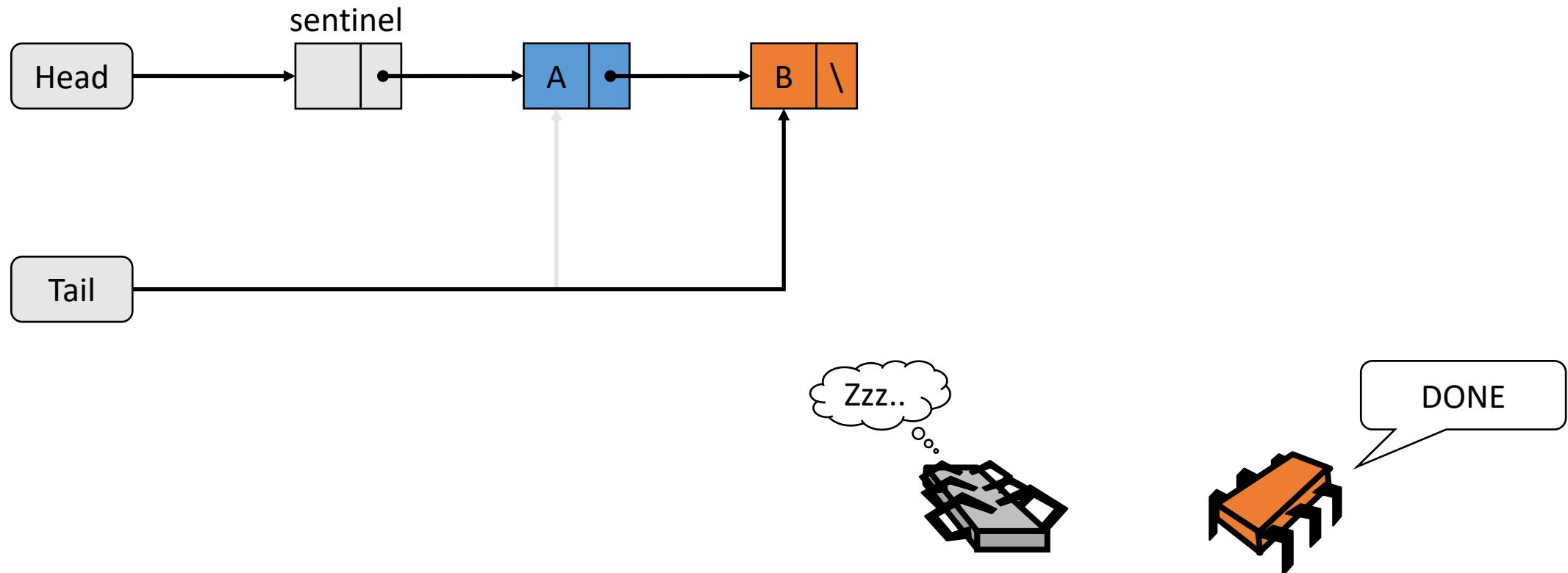
# Unbounded Lock-free Queue



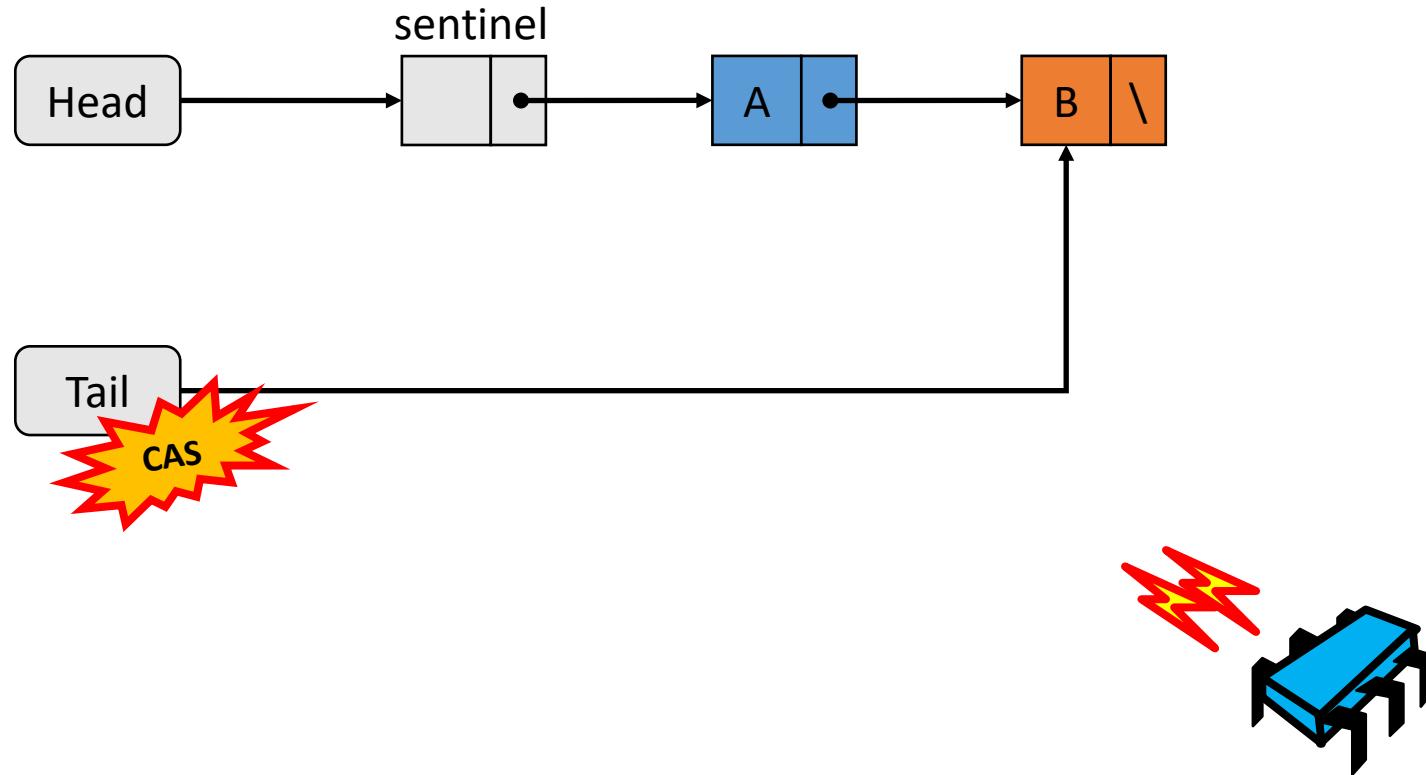
# Unbounded Lock-free Queue



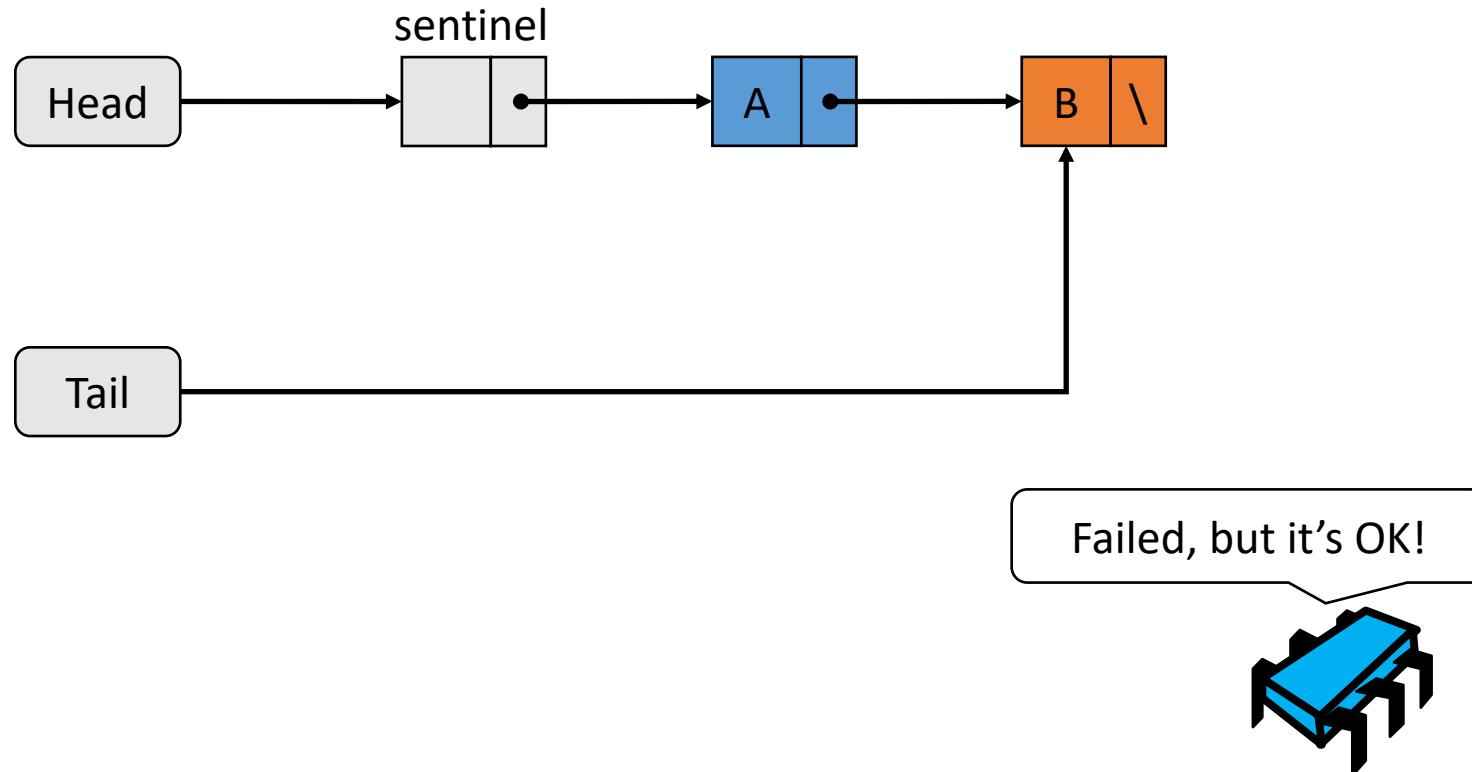
# Unbounded Lock-free Queue



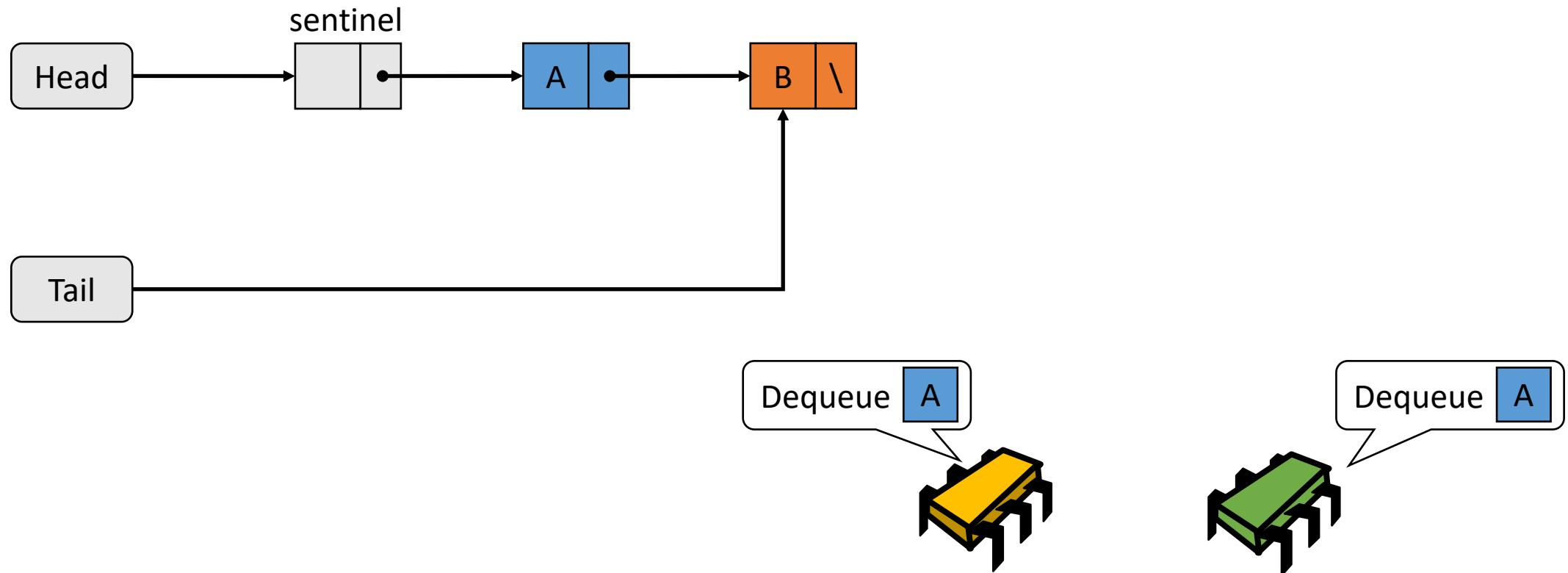
# Unbounded Lock-free Queue



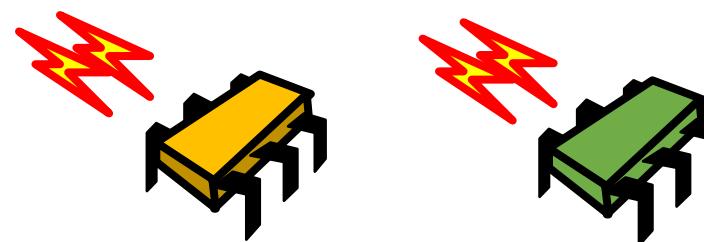
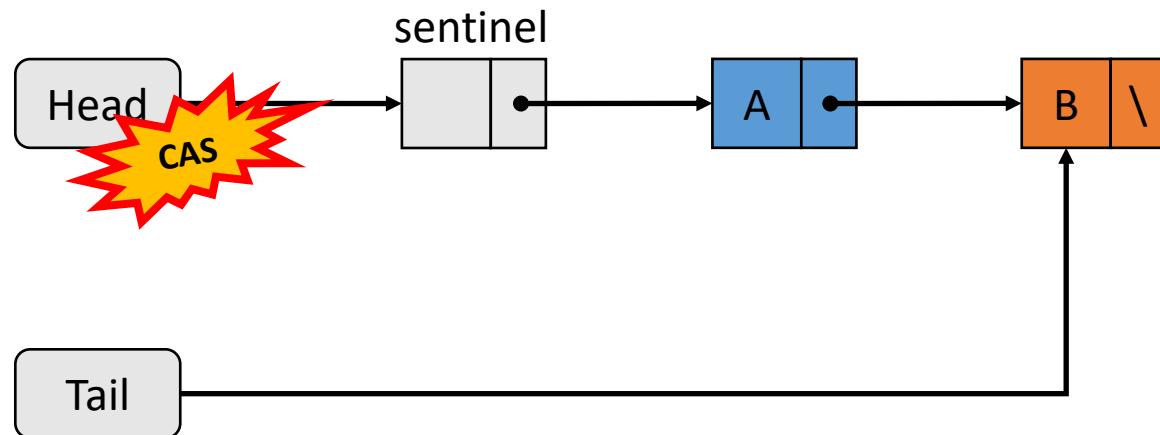
# Unbounded Lock-free Queue



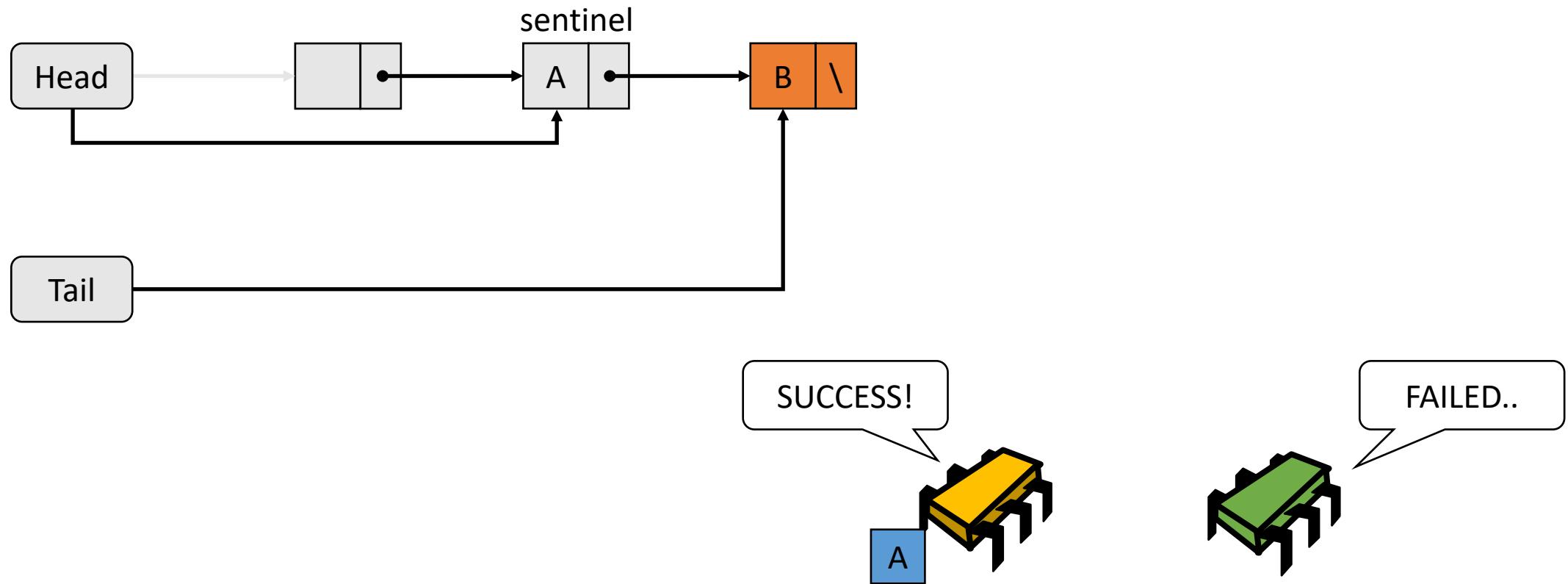
# Unbounded Lock-free Queue



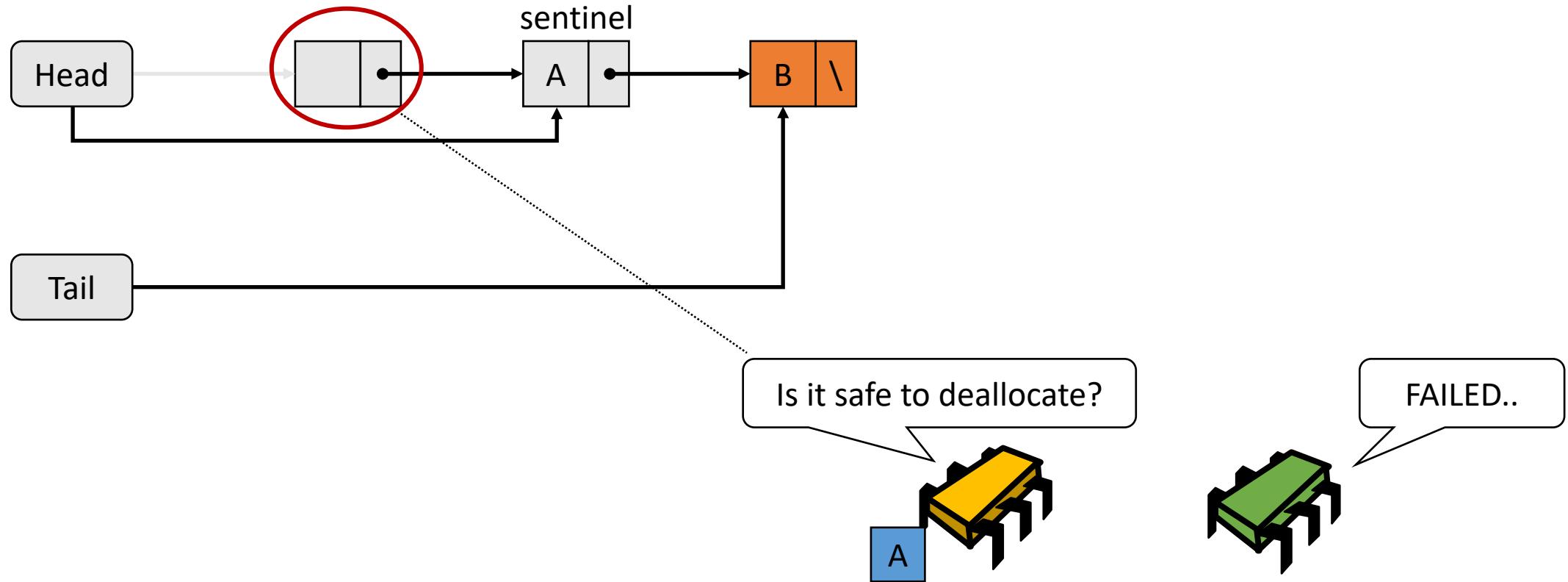
# Unbounded Lock-free Queue



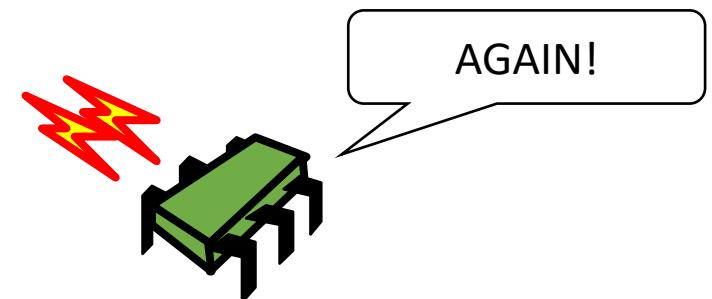
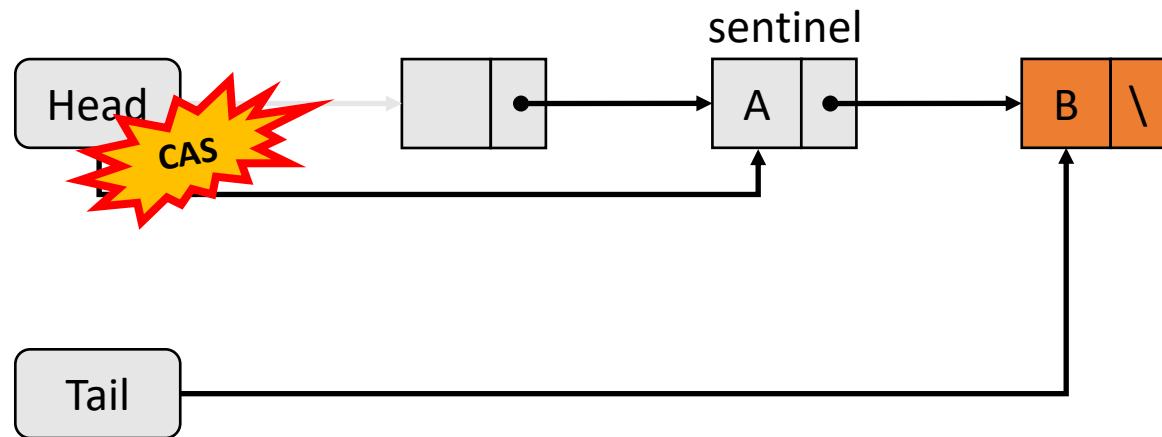
# Unbounded Lock-free Queue



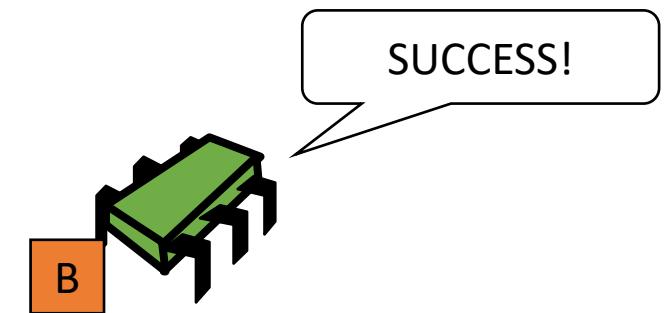
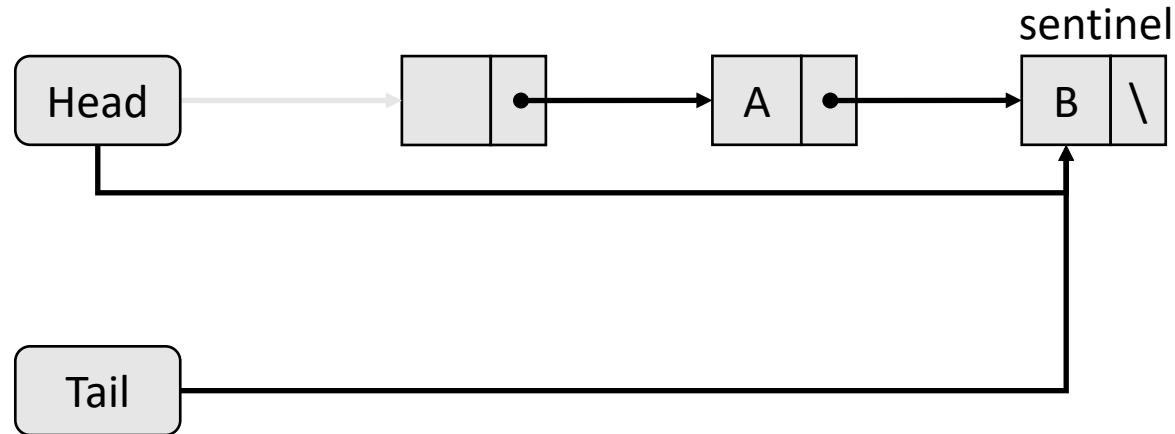
# Unbounded Lock-free Queue



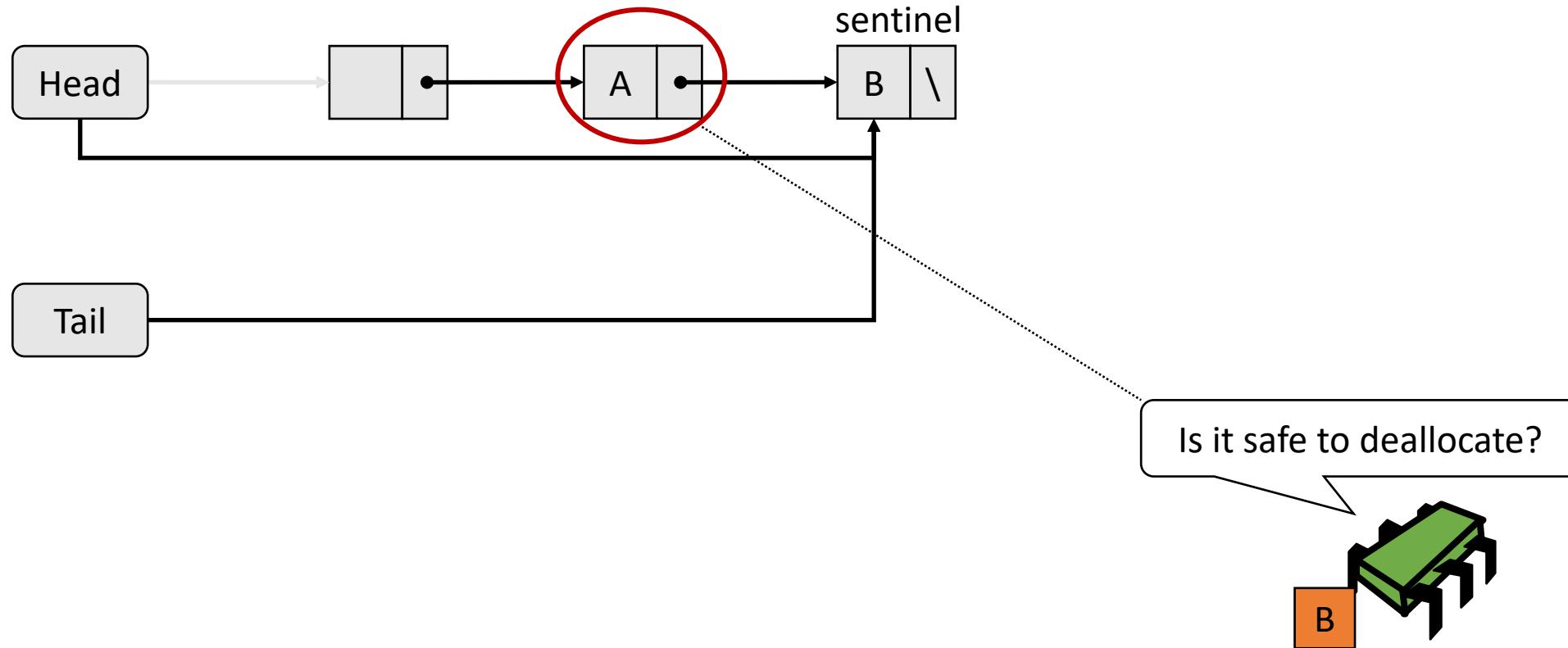
# Unbounded Lock-free Queue



# Unbounded Lock-free Queue



# Unbounded Lock-free Queue



# Evaluation

CPU: 24 x 2 (Hyperthreading enabled)

Number of producer / consumer threads: 16

Number of enqueue / dequeue operations per thread: 1,000,000

```
[jongbin@multicore-24:~/TA/Multicore/lab13$ time ./queue_giantlock
CORRECT!

real    0m5.593s
user    0m4.432s
sys     2m49.276s
```

```
[jongbin@multicore-24:~/TA/Multicore/lab13$ time ./queue_unbounded
CORRECT!

real    0m14.373s
user    7m3.264s
sys     0m8.220s
```

```
[jongbin@multicore-24:~/TA/Multicore/lab13$ time ./queue_bounded
CORRECT!

real    0m1.972s
user    0m56.572s
sys     0m0.848s
```

# Thank You