

1

- **synchronized**  Lock 

1

- 
  - 

CAS

## java.util.concurrent [ ] CAS [ ]

CAS

-  V
  -  A
  -  B

1

- **V**A B
  -

- 

1

A decorative horizontal bar consisting of a series of small, evenly spaced rectangles.

- 
  - 

1

```
Lock l = new ReentrantLock(true);
```

1

- 
  - 

2

```
Lock l = new ReentrantLock(false);
```

1

- synchronized  $\sqcap$  ReentrantReadWriteLock.WriteLock  $\sqcap$  Lock  $\sqcap\sqcap\sqcap\sqcap\sqcap\sqcap$

A horizontal row of seven empty rectangular boxes, likely intended for a child to draw or write in.

A horizontal row of fifteen empty rectangular boxes, intended for children to write their names in, likely as part of a classroom activity.

- ReentrantReadWriteLock
  - ReentrantReadWriteLock.ReadLock

\_\_\_\_\_

- ReentrantLock  $\sqsubset$  synchronized 互斥锁
  - 顺序一致性

## Synchronize

- 
  - 

11

A decorative horizontal bar consisting of a series of small, evenly spaced rectangular boxes, likely a separator or a decorative element at the bottom of the page.

- CAS

三

A horizontal row of 20 small, empty rectangular boxes, likely used for input fields or placeholder text in a form.

- CAS
  - CAS
  - - CAS
    - JVM
  - CAS

1

- **Java Virtual Machine** JVM Mark Word
    - JVM CAS Mark Word ID
    - Java Object Mark Word
  - **Object** Mark Word ID

1

-  CAS

5 / 5

- JVM上锁操作通过调用`lock`方法实现，该方法会将当前线程的`Mark Word`写入到`Lock Record`中。
  - JVM使用CAS操作\*\*`Mark Word`\*\*来检测`Lock Record`是否被其他线程修改。如果修改，则表示`Lock Record`的`owner`不是当前线程，`Mark Word`也非\*\*`Mark Word`\*\*。
  - 如果`Mark Word`等于`Mark Word`，则表示`Lock Record`未被其他线程修改，从而完成上锁操作。
  - 上锁成功后，JVM会将`Mark Word`设置为`LOCKED`状态。
    - 通过`LOCKED`状态的`Mark Word`，其他线程可以知道该线程正在持有锁。
    - 通过`LOCKED`状态的`Mark Word`，其他线程可以知道该线程持有的锁是公平锁还是偏向锁。

4

## Mutex

- **ObjectMonitor**
  - **C++ 用の ObjectMonitor 実装**
    - **ObjectMonitor.h**
    - **ObjectMonitor.cpp**
      - **Object**
      - **ObjectTable**

4



## ObjectMonitor

1. `如果当前线程的 _recursions==0 则直接返回`
  2. `将当前线程加入 _EntryList 队列中`
- `如果当前线程的 _WaitSet 不为空`
    1. `从 _WaitSet 中移除该线程`
    2. `调用 notify() 方法将该线程加入 _EntryList 队列中`