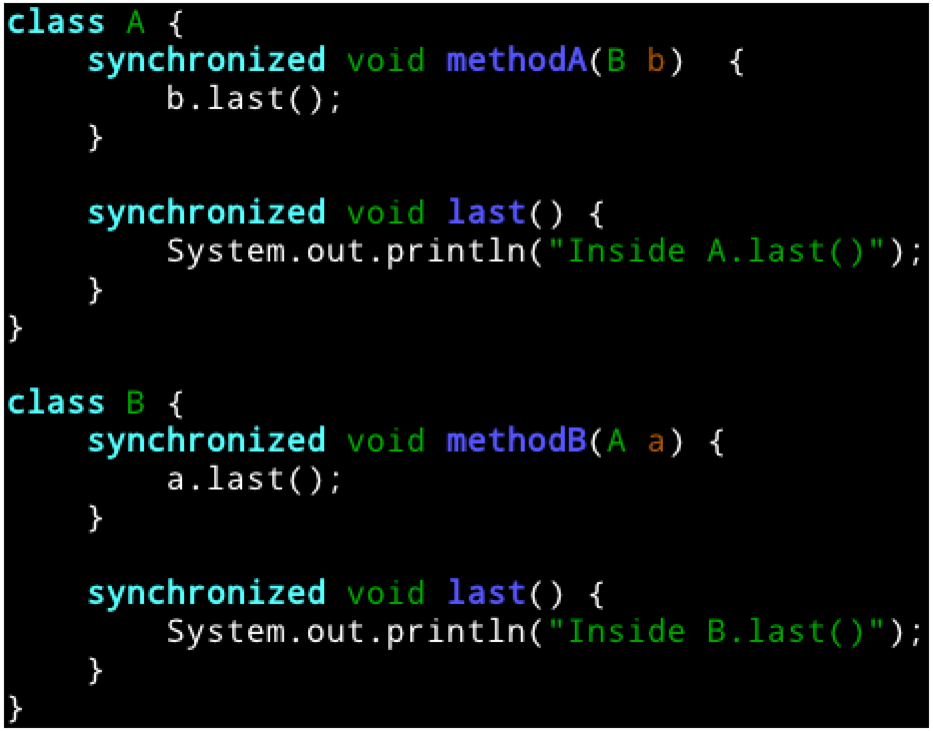
中山大學 计算机学院（软件学院）

SUN YAT-SEN UNIVERSITY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Replicable deadlock

Kai Huang

A Replicable Simple Example

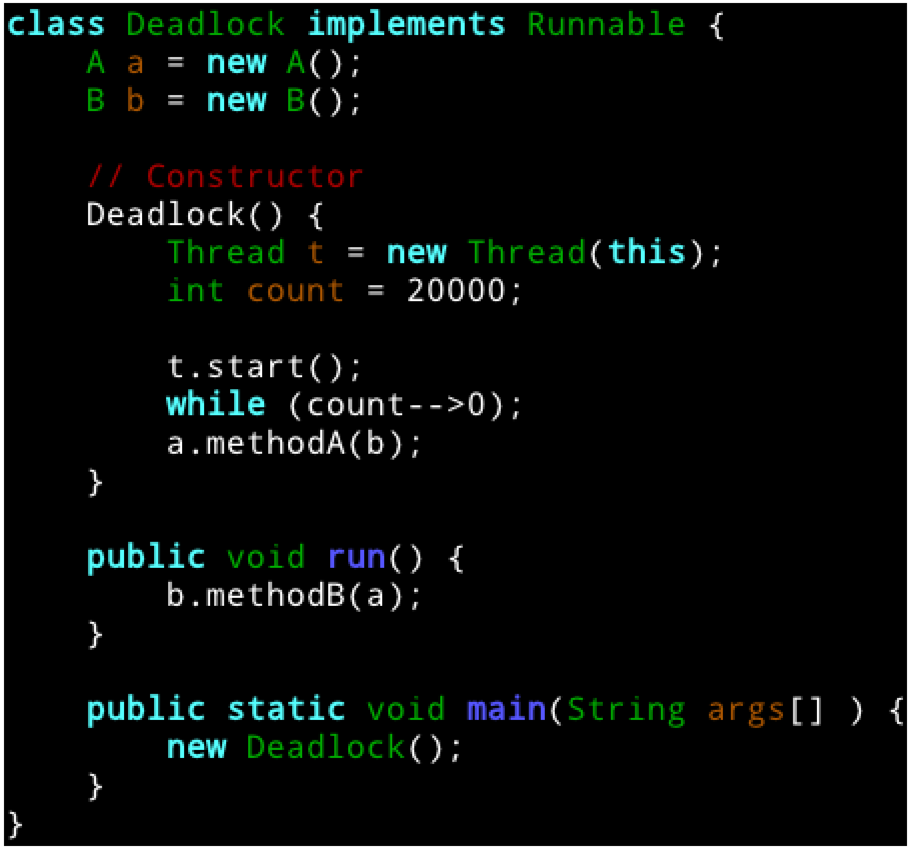


```java
class A {
    synchronized void methodA(B b)  {
        b.last();
    }

    synchronized void last() {
        System.out.println("Inside A.last()");
    }
}

class B {
    synchronized void methodB(A a) {
        a.last();
    }

    synchronized void last() {
        System.out.println("Inside B.last()");
    }
}
```



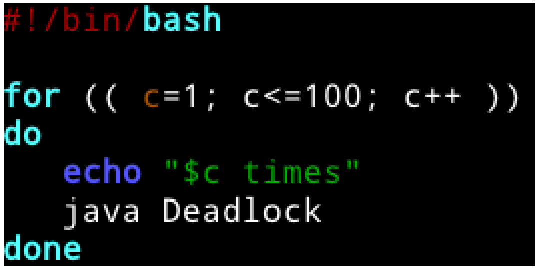
```java
class Deadlock implements Runnable {
    A a = new A();
    B b = new B();

    // Constructor
    Deadlock() {
        Thread t = new Thread(this);
        int count = 20000;

        t.start();
        while (count-->0);
        a.methodA(b);
    }

    public void run() {
        b.methodB(a);
    }

    public static void main(String args[] ) {
        new Deadlock();
    }
}
```



```bash
#!/bin/bash

for (( c=1; c<=100; c++ ))
do
   echo "$c times"
   java Deadlock
done
```

Another article:
http://javaeesupportpatterns.blogspot.ca/2013/01/java-concurrency-hidden-thread-deadlocks.html

要求

- 在三个不同平台运行上述程序
  - Linux vs. Windows
  - Desktop vs. Laptop
  - PC vs. Mobile device