

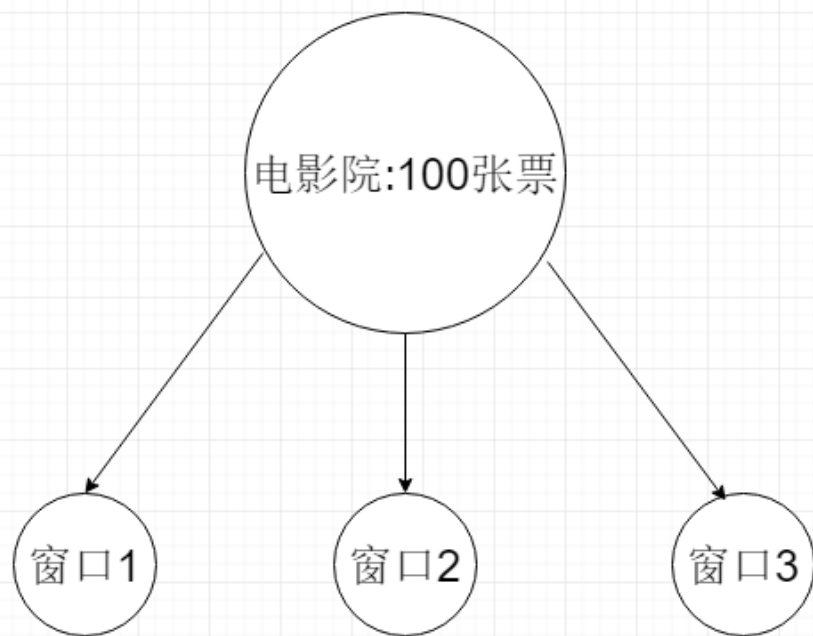
线程和线程池

进程和线程

- 进程:一个启动的应用程序,是线程的载体
- 线程是系统中最小的执行单元
- 启动引用程序启动进程,进程默认情况下会启动主线程

线程创建

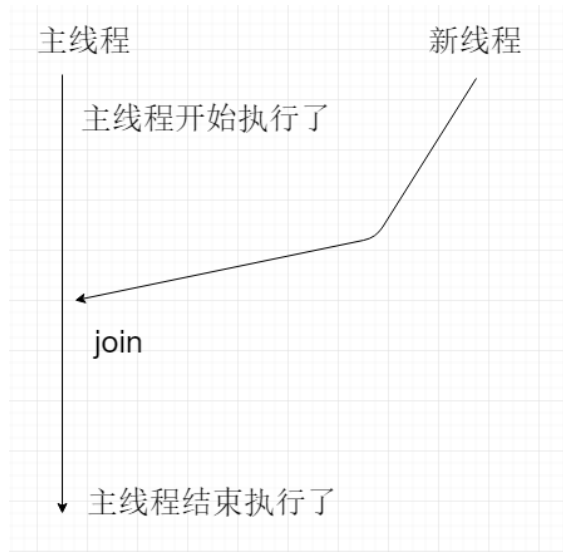
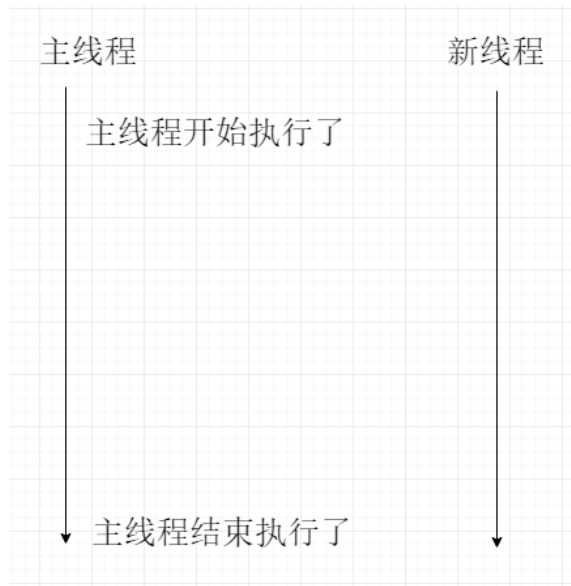
- 直接通过Thread
- 通过Runnable



线程join

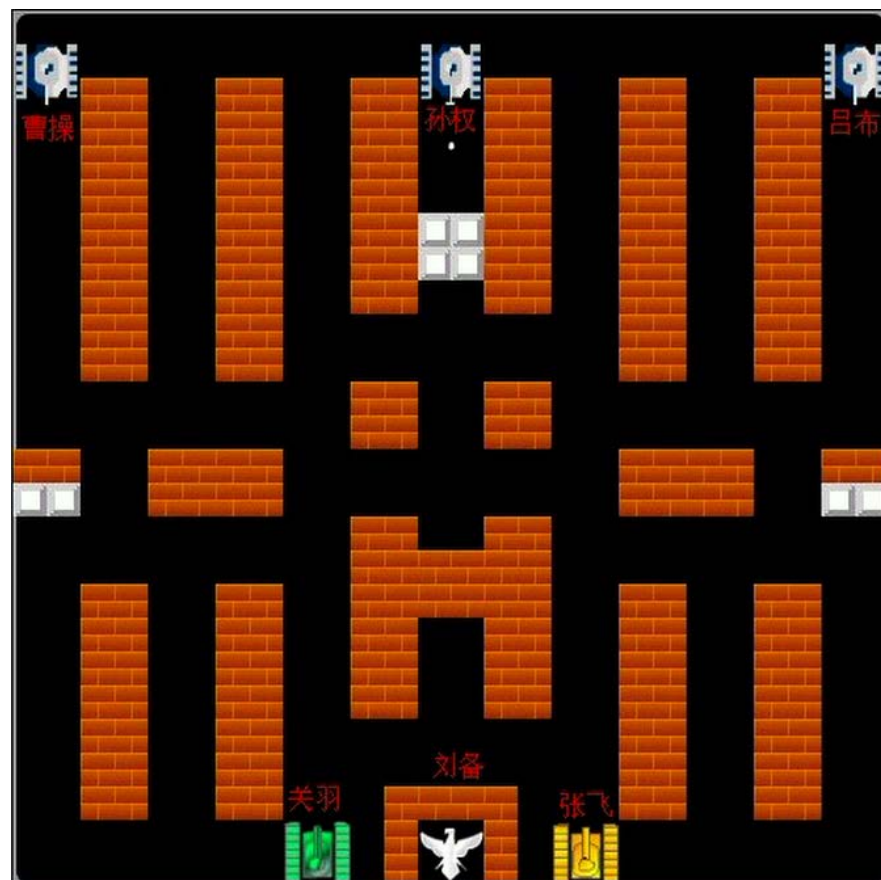
- 将线程之间的并行执行变为串行执行

线程join



守护线程

- 用户线程和守护线程
- 守护线程是指在程序运行的时候在后台提供一种通用的服务的线程(垃圾回收线程)
- 主线程执行结束,守护线程就结束



线程池

- 程序启动一个线程成本是比较高的,因为它涉及到要与操作系统进行交互.而使用线程池可以很好的提高性能,尤其是当程序中要创建大量生命期很短的线程时,更应该考虑使用线程池.
- 线程池里的每一个线程代码结束后,并不会死亡,而是再次回到线程池中成为空闲状态,等待下一个对象来使用
- 在JDK5之前,我们需要自己手动实现自己的线程池,从JDK5开始,java内置支持线程池

线程池

//第一步:创建线程池

```
ExecutorService service = Executors.newFixedThreadPool( nThreads: 2 );
```

//第二步:创建Runnable对象

```
MyRunnable runnable = new MyRunnable();
```

//第三步:执行线程

```
service.execute(runnable);
```

