**GPU分配测试**

一个GPU只能够分配给一个实例，如果一台机器只有一个GPU，首先给一个实例分配了一个GPU，如果这个实例处于运行过程中（运行过程指实例notebook server处在运行状态），再给另一个实例分配GPU的时候会一直卡在创建实例的步骤。因此一个GPU只能够分配给一个实例。如果停止占用GPU的实例，再创建需要分配GPU的实例就可以分配成功。**总结：GPU只能被一个运行中的实例使用。**

**内存分配测试**

一个模块分配了指定量的内存之后不会单独占有这一部分内存，但是应该是会声明自己占有了多大的内存让kubeflow知道，当kubeflow在创建新的实例，并为其分配内存的时候会考虑到这一点，当（总内存-当前宿主机已占用的内存-其他模块声明的内存）< 需要的额内存的时候会一直卡在实例创建的状态。推测kubeflow会避免内存申请超过总内存的情况和实际可用内存的情况，以避免出现问题。**总结：kubeflow分配的内存不会大于总内存，且内存不会重复分配。**

由于资源的分配策略分配的资源不会重复，因此资源不会产生竞争情况。