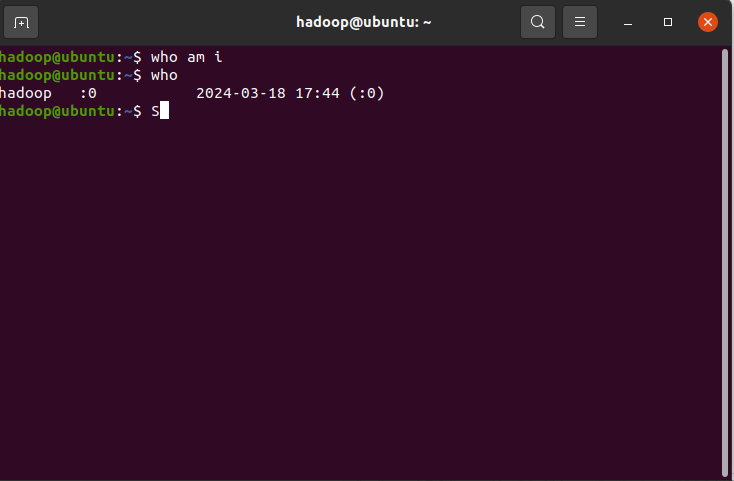
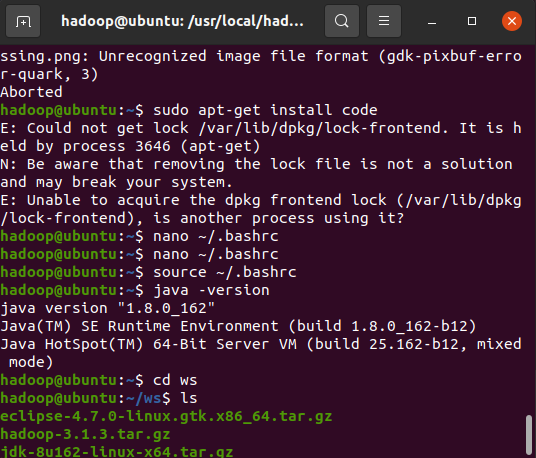
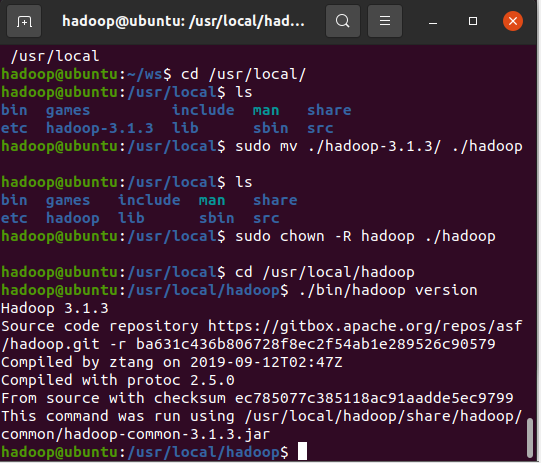
## 前置环境配置

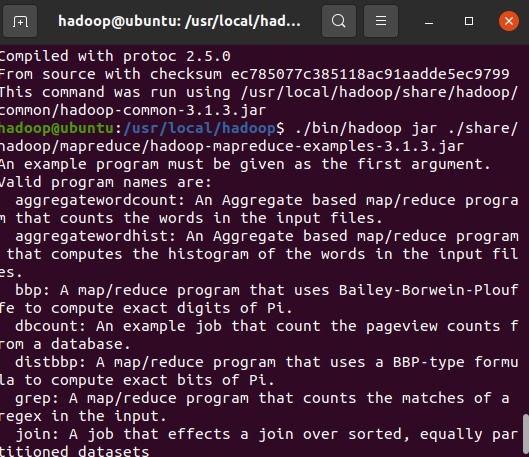
查看当前用户名正确  
  
#VMWARE  
VMWARE 鼠标闪烁问题需要设置鼠标为拖尾运行即可



* 安装jvm成功  
    
  安装hadoop成功

## hadoop单机配置

./bin/hadoop jar ./share/hadoop/mapreduce/hadoop-mapreduce-examples-3.1.3.jar  
复制

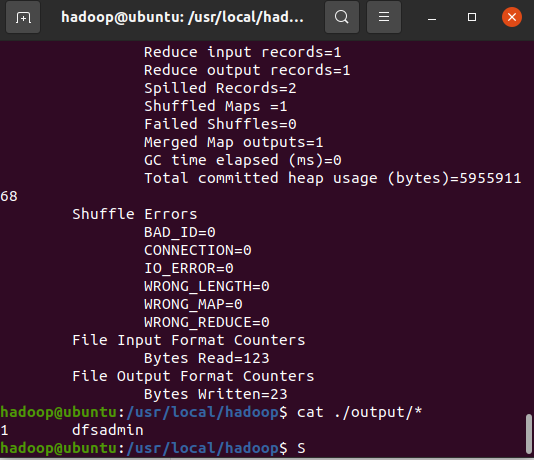
执行后可以看到所有的示例程序  


测试样例

运行指令

在此我们选择运行 grep 例子，我们将 input 文件夹中的所有文件作为输入，筛选当中符合正则表达式 dfs[a-z.]+ 的单词并统计出现的次数，最后输出结果到 output 文件夹中。

cd /usr/local/hadoop  
mkdir ./input  
cp ./etc/hadoop/\*.xml ./input # 将配置文件作为输入文件  
./bin/hadoop jar ./share/hadoop/mapreduce/hadoop-mapreduce-examples-3.1.3.jar grep ./input ./output 'dfs[a-z.]+'  
cat ./output/\* # 查看运行结果  
复制

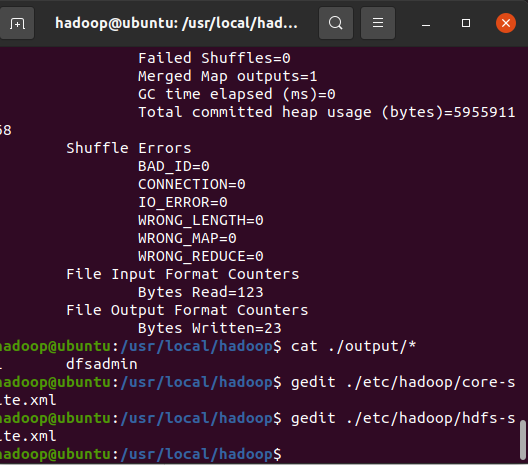


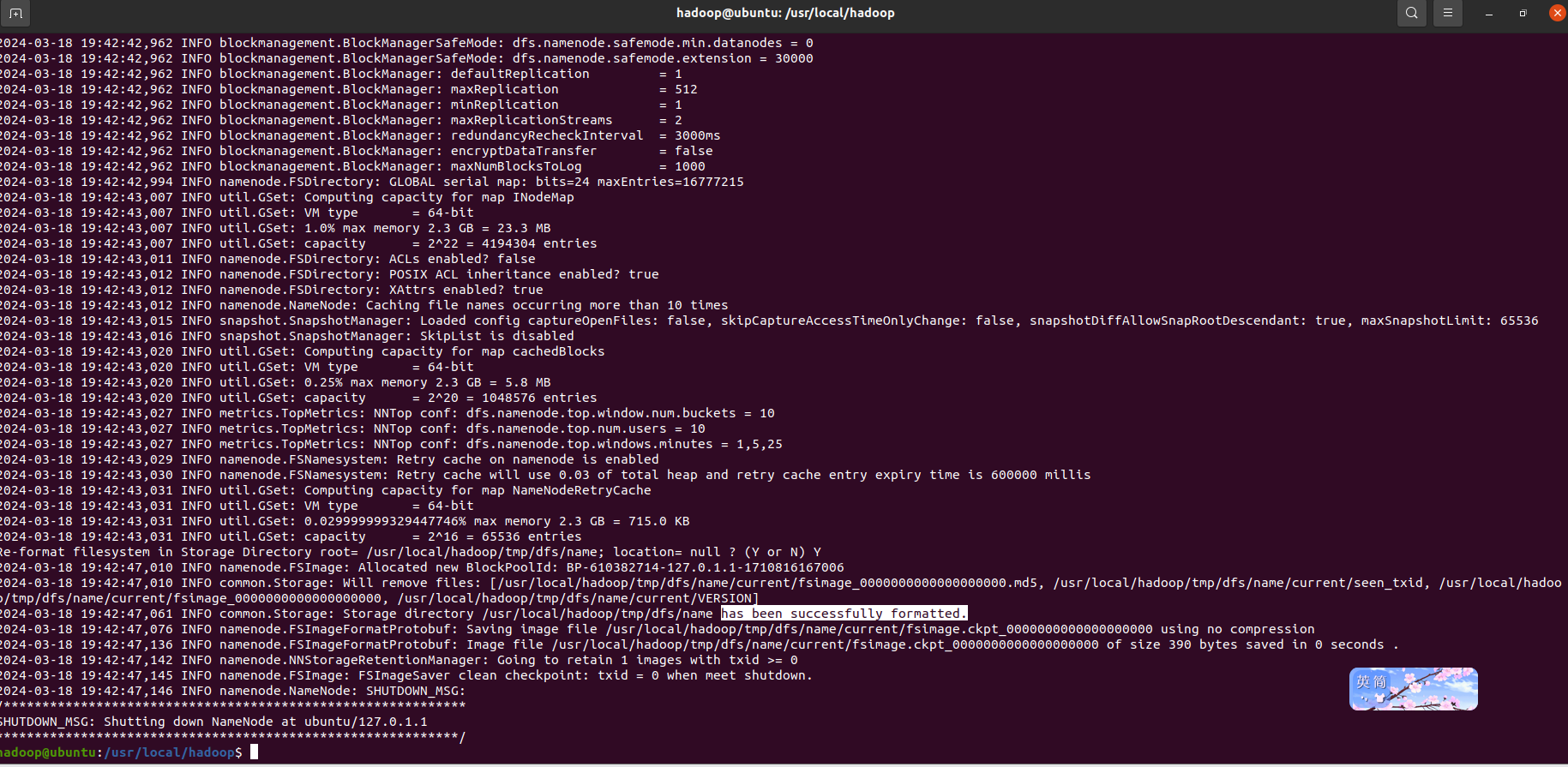
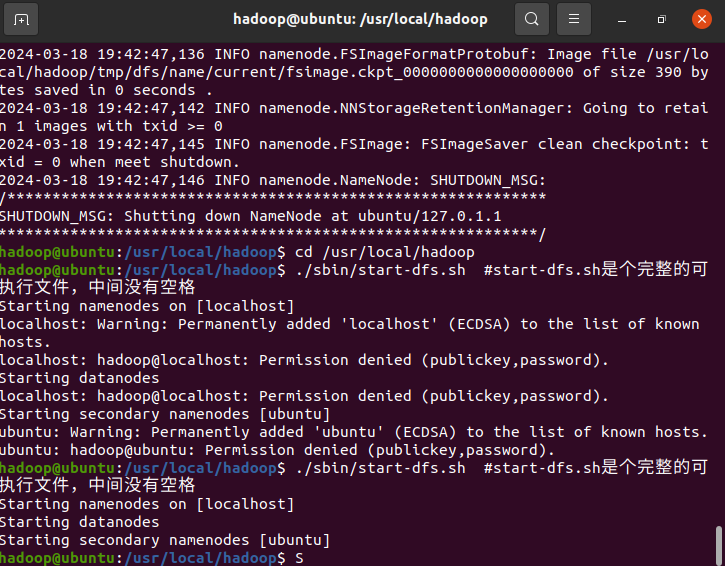
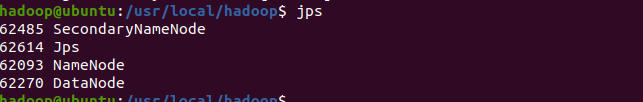
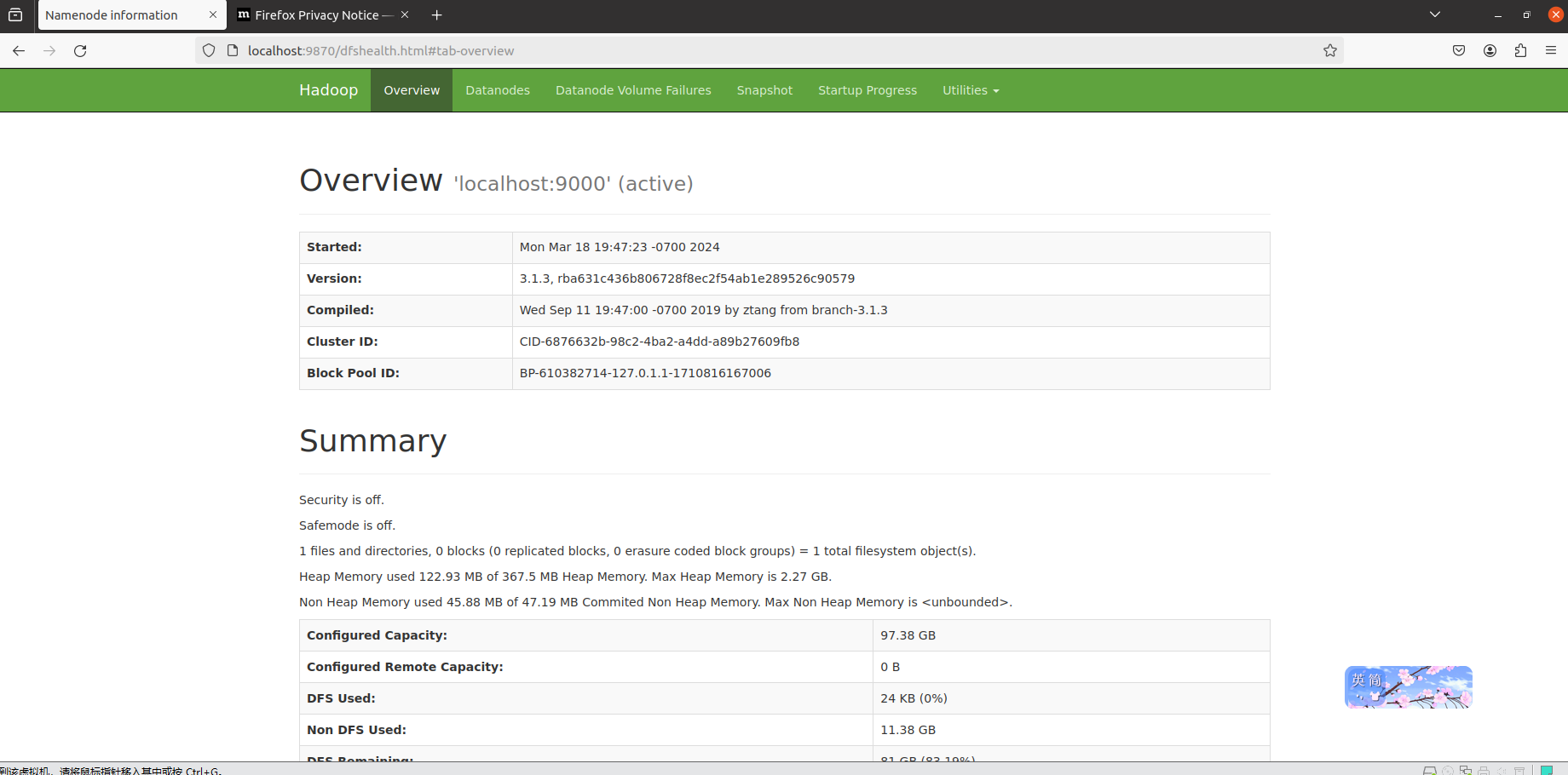
执行成功后，输出了作业的相关信息，输出的结果是符合正则的单词 dfsadmin 出现了1次

## hadoop伪分布式配置

Hadoop 可以在单节点上以伪分布式的方式运行，Hadoop 进程以分离的 Java 进程来运行，节点既作为 NameNode 也作为 DataNode，同时，读取的是 HDFS 中的文件。

Hadoop 的配置文件位于 /usr/local/hadoop/etc/hadoop/ 中，伪分布式需要修改2个配置文件 **core-site.xml** 和 **hdfs-site.xml** 。Hadoop的配置文件是 xml 格式，每个配置以声明 property 的 name 和 value 的方式来实现。

伪分布式虽然只需要配置 fs.defaultFS 和 dfs.replication 就可以运行（官方教程如此），不过若没有配置 hadoop.tmp.dir 参数，则默认使用的临时目录为 /tmp/hadoo-hadoop，而这个目录在重启时有可能被系统清理掉，导致必须重新执行 format 才行。

  
formate 成功  
接着开启 NameNode 和 DataNode 守护进程。  
  
出现了一个关于 ssh permission denied 错误  
通过重装ssh授权解决  
正确启动 jps命令验证通过  
  
网页看板启动正常  
http://localhost:9870/dfshealth.html#tab-overview  


## 运行Hadoop 伪分布式实例

上面的单机模式，grep 例子读取的是本地数据，伪分布式读取的则是 HDFS 上的数据。要使用 HDFS，首先需要在 HDFS 中创建用户目录：

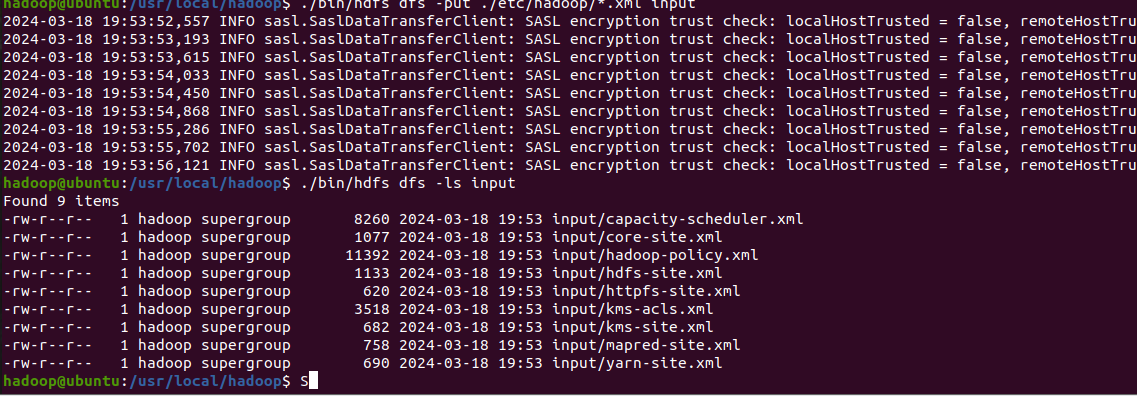
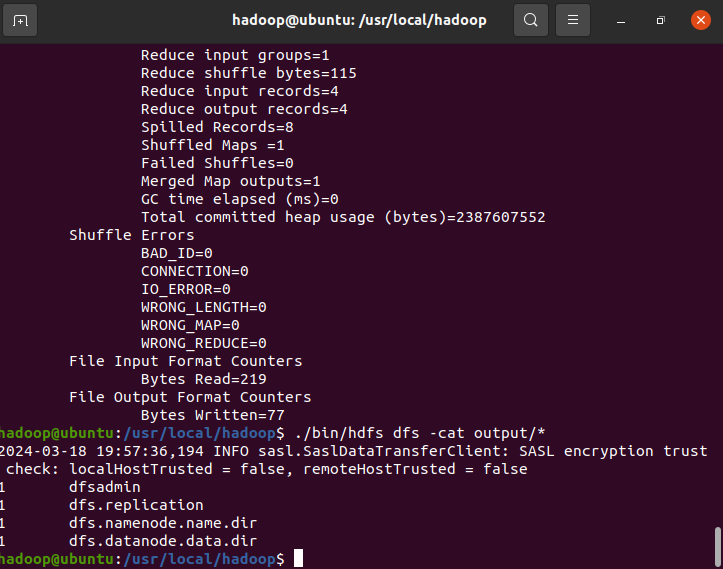
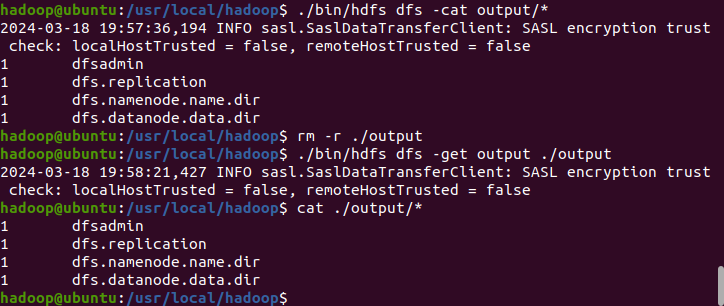
注意 有三种shell命令方式。

1. hadoop fs
2. hadoop dfs
3. hdfs dfs

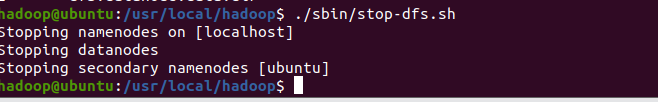
hadoop fs适用于任何不同的文件系统，比如本地文件系统和HDFS文件系统  
hadoop dfs只能适用于HDFS文件系统  
hdfs dfs跟hadoop dfs的命令作用一样，也只能适用于HDFS文件系统

1. 复制  ./etc/hadoop 中的 xml 文件作为输入文件复制到分布式文件系统中

./bin/hdfs dfs -mkdir input  
./bin/hdfs dfs -put ./etc/hadoop/\*.xml input  
复制

  
  
Hadoop 运行程序时，输出目录不能存在，否则会提示错误 "org.apache.hadoop.mapred.FileAlreadyExistsException: Output directory hdfs://localhost:9000/user/hadoop/output already exists" ，因此若要再次执行，需要执行如下命令删除 output 文件夹:  
  
实际开发时要在代码里加入以下代码确保运行过程中不会因为删除文件的问题报错

Configuration conf = new Configuration();  
Job job = new Job(conf);  
   
/\* 删除输出目录 \*/  
Path outputPath = new Path(args[1]);  
outputPath.getFileSystem(conf).delete(outputPath, true);  
复制

若要关闭 Hadoop，则运行  
*注意*: 下次启动 hadoop 时，无需进行 NameNode 的初始化，只需要运行 ./sbin/start-dfs.sh 就可以！