实　　验　　报　　告

学　　号	1902001 1038	姓　　名	岳宇轩	专业班级	2019 级计算机科学与技术慧与卓越工程师班
课程名称	**大数据导论**			学期	**2022 年秋季学期**
任课教师	刘洁 刘艳艳	完成日期	2022.11.07	上机课时间	周一 56 节（双周）
实 验 名 称	实验 5 MapReduce 编程初级实践				

一、实验要求（10%）

1. 实验目的

1.通过实验掌握基本的 MapReduce 编程方法；

2.掌握用 MapReduce 解决一些常见的数据处理问题，包括数据去重、数据排序和数据挖掘等。

2. 实验平台

已经配置完成的 Hadoop 伪分布式环境。

参考以下环境配置：

Ubuntu 下 Hadoop 伪分布式环境配置：http://dblab.xmu.edu.cn/blog/install-hadoop-in-centos/

Ubuntu 下使用 Eclipse 编译运行 MapReduce 程序示例：
http://dblab.xmu.edu.cn/blog/hadoop-build-project-using-eclipse/

二、实验内容及步骤（80%）

1.编程实现文件合并和去重操作

对于两个输入文件，即文件 A 和文件 B，请编写 MapReduce 程序，对两个文件进行合并，并剔除其中重复的内容，得到一个新的输出文件 C。下面是输入文件和输出文件的一个样例供参考。

输入文件 A 的样例如下：

```
20150101    x
20150102    y
20150103    x
20150104    y
20150105    z
20150106    x
```

输入文件 B 的样例如下：

```
20150101    y
```

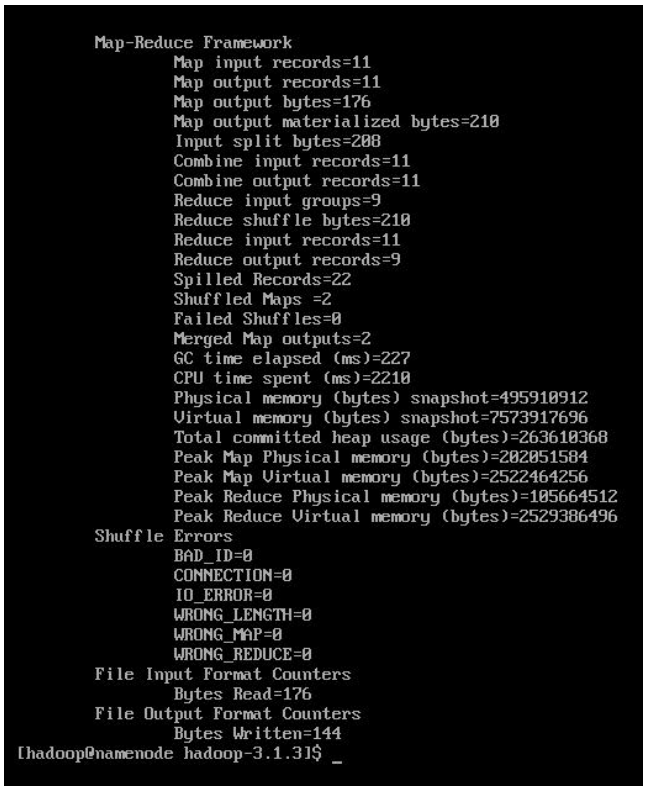
```
20150102    y
20150103    x
20150104    z
20150105    y
```

根据输入文件 A 和 B 合并得到的输出文件 C 的样例如下：

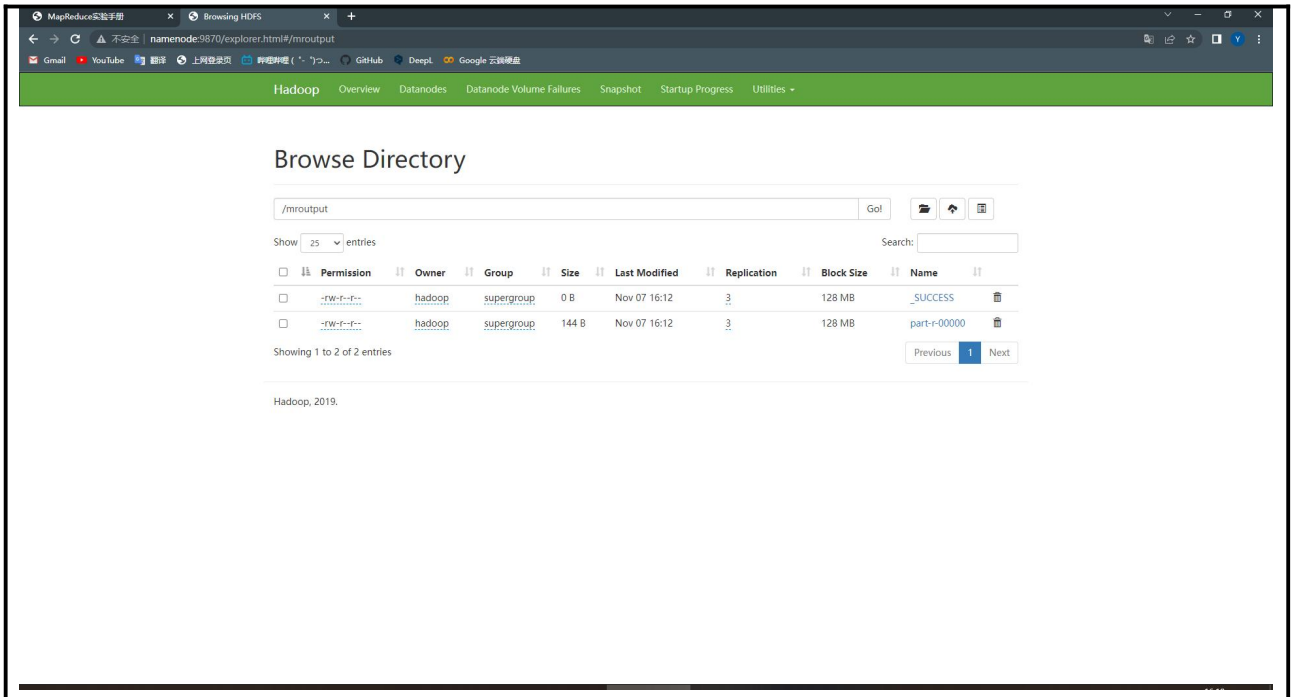
```
20150101    x
20150101    y
20150102    y
20150103    x
20150104    y
20150104    z
20150105    y
    20150105    z
20150106    x
```

实验结果：

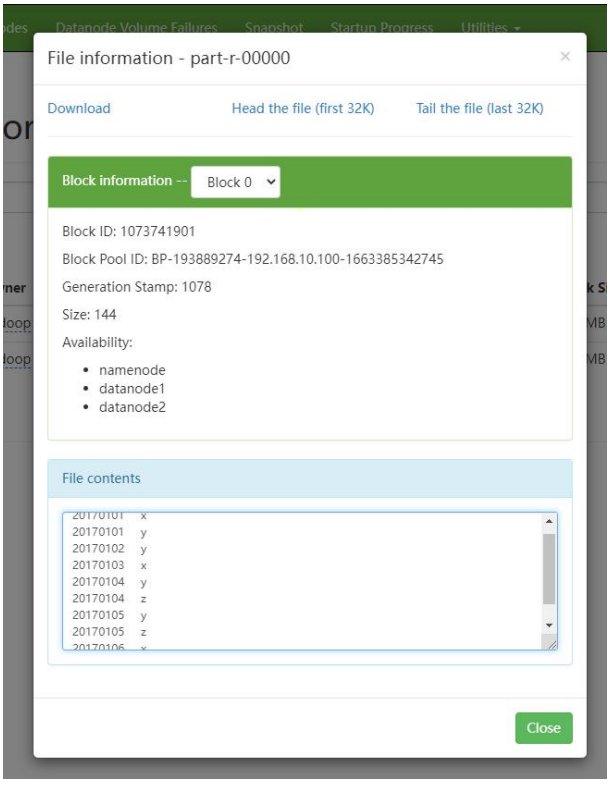
在 namenode 上运行 jar 包之后，得到如下显示：



进入 hdfs 的 web 端页面查看，可以看到 mroutput 中有两个文件：



点开第二个文件进行预览，可以看到输出与样例一致：



2. 编写程序实现对输入文件的排序

现在有多个输入文件，每个文件中的每行内容均为一个整数。要求读取所有文件中的整数，进行升序排序后，输出到一个新的文件中，输出的数据格式为每行两个整数，第一个数字为第二个整数的排序位次，第二个整数为原待排列的整数。下面是输入文件和输出文件的一个样例供参考。

输入文件 1 的样例如下：

```
33
37
12
40
```

输入文件 2 的样例如下：

```
4
16
39
5
```

输入文件 3 的样例如下：

```
1
45
25
```

根据输入文件 1、2 和 3 得到的输出文件如下：

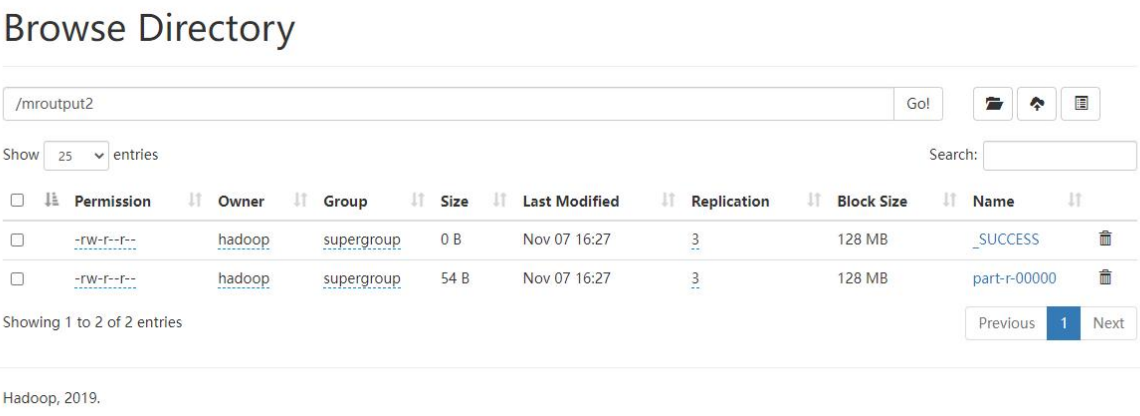
```
1 1
2 4
3 5
4 12
5 16
6 25
7 33
8 37
9 39
10 40
11 45
```

实验结果：

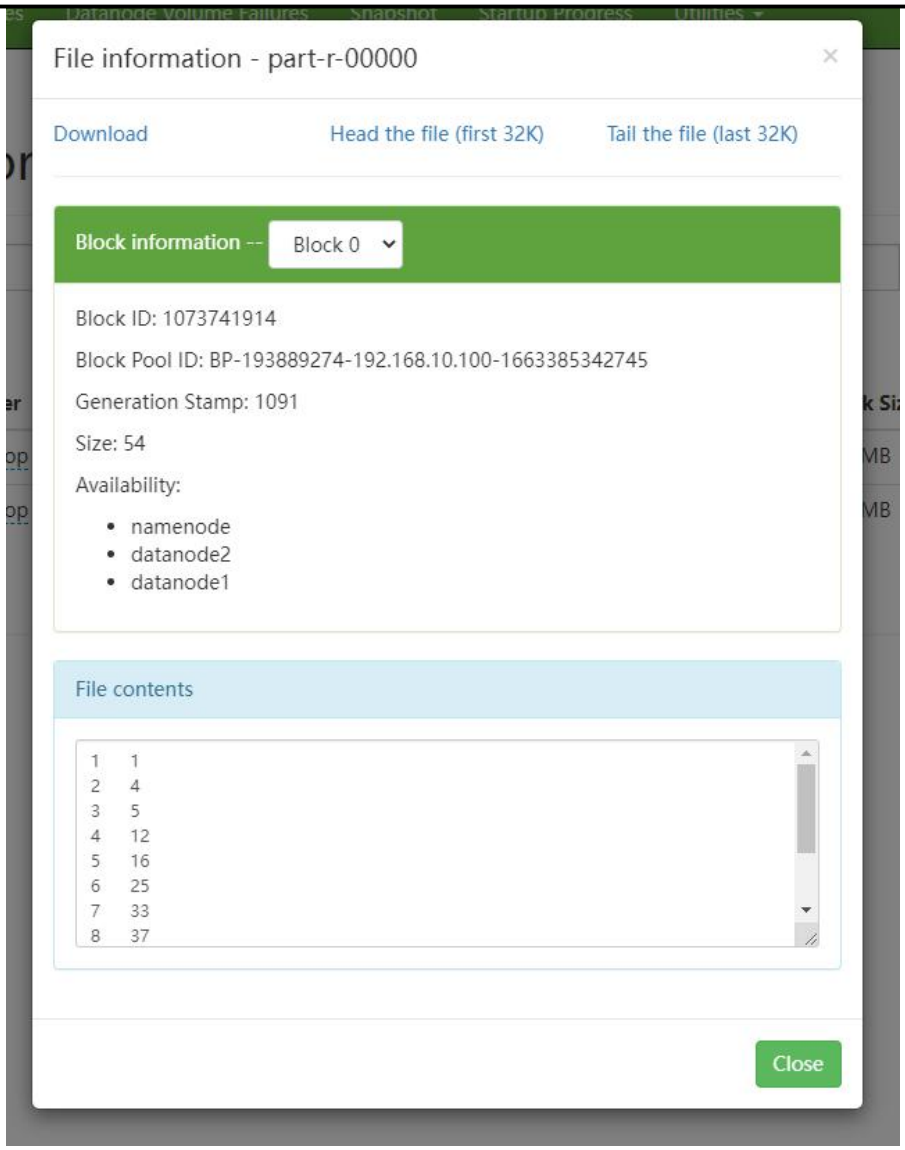
在 namenode 上运行 jar 包之后，得到如下显示：

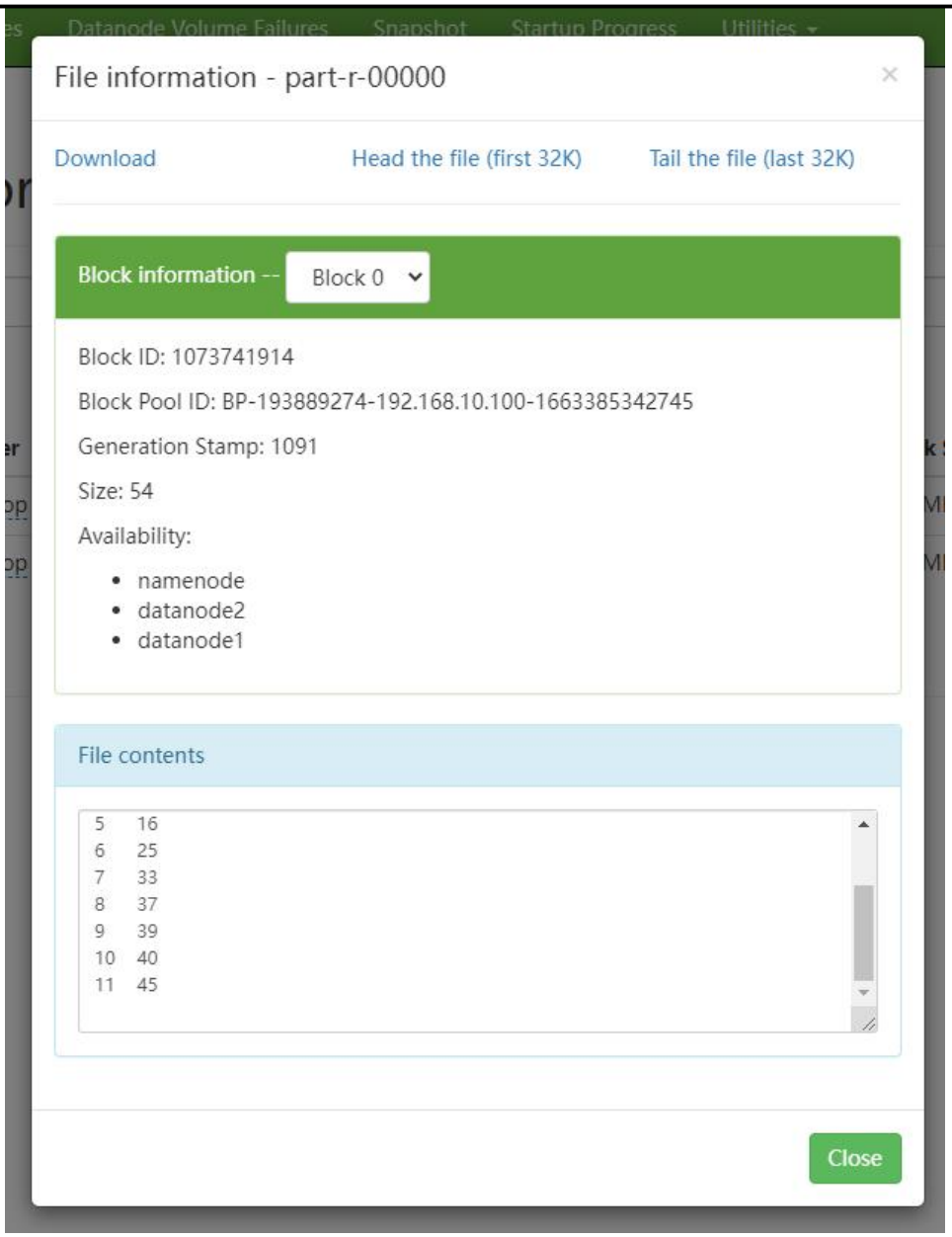
```
        Map-Reduce Framework
                Map input records=11
                Map output records=11
                Map output bytes=88
                Map output materialized bytes=128
                Input split bytes=315
                Combine input records=0
                Combine output records=0
                Reduce input groups=11
                Reduce shuffle bytes=128
                Reduce input records=11
                Reduce output records=11
                Spilled Records=22
                Shuffled Maps =3
                Failed Shuffles=0
                Merged Map outputs=3
                GC time elapsed (ms)=265
                CPU time spent (ms)=1730
                Physical memory (bytes) snapshot=690368512
                Virtual memory (bytes) snapshot=10096074752
                Total committed heap usage (bytes)=391176192
                Peak Map Physical memory (bytes)=203472896
                Peak Map Virtual memory (bytes)=2522464256
                Peak Reduce Physical memory (bytes)=110489600
                Peak Reduce Virtual memory (bytes)=2529476608
        Shuffle Errors
                BAD_ID=0
                CONNECTION=0
                IO_ERROR=0
                WRONG_LENGTH=0
                WRONG_MAP=0
                WRONG_REDUCE=0
        File Input Format Counters
                Bytes Read=35
        File Output Format Counters
                Bytes Written=54
[hadoop@namenode hadoop-3.1.3]$
```

进入 hdfs 的 web 端页面查看，可以看到 mroutput2 中有两个文件：



点开第二个文件进行预览，可以看到输出与样例一致：


File information - part-r-00000
Download
Head the file (first 32K)
Tail the file (last 32K)
Block information -- Block 0
Block ID: 1073741914
Block Pool ID: BP-193889274-192.168.10.100-1663385342745
Generation Stamp: 1091
Size: 54
Availability:
namenode
datanode2
datanode1
File contents
1 1
2 4
3 5
4 12
5 16
6 25
7 33
8 37
Close



三、心得总结（写出自己在完成实验过程中遇到的问题、解决方法，以及体会、收获等）（10%）

通过本次实验，我了解了 MapReduce 的安装和配置过程。在使用 IDEA 进行本地编译的时候，我查看 pom.xml 文件中的如下两行会标红。在 hdfs 的实验中我也遇到过同样的问题。这次我通过在 artifactionId 前加入 groupId 解决了标红的问题。

```
    <artifactId>maven-compiler-plugin</artifactId>
    <version>3.6.1</version>
    <configuration>
        <source>1.8</source>
        <target>1.8</target>
    </configuration>
</plugin>
<plugin>
    <artifactId>maven-assembly-plugin</artifactId>
    <configuration>
```

之后，我总是遇到程序退出并报错 exit with code 1。我查看控制台输出的内容是在第 67 行提交 job 时报错：

```
66          // 7 提交job
67          System.exit(job.waitForCompletion(true) ? 0 : 1);
```

在这里我卡了有接近两个小时，尝试了各种办法。一开始我以为这一句是写错了，便查了一下，得知这一句是 mapreduce 中常用的判断短程序是否退出的方法，没有错。

我按照助教给的说明文件从头排查，我以为是我用的 3.8.6 版本的 maven 可能会出问题，于是换成了助教给的 3.5.4 又配了一遍，发现还是不行。

最后在网上翻了好多博客，有的说可能是中文路径的问题，（我原来是直接用 IDEA 打开“MapReduce 软件包”作为一个 project 的）。所以我从新建项目开始一步一步的构建了一个项目，最后成功运行。