《并行计算》上机报告

姓名：	裴启智	学号：	PB18111793	日期：	2021.6.9
上机题目：	Hadoop 实验				

实验环境：
CPU：Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz 2.21 GHz；
内存: 16GB 操作系统：Windows WSL2 软件平台：vscode

一、算法设计与分析：

题目一：
按照 Hadoop 安装运行说明文档中的指导自己搭建伪分布式 Hadoop 环境，熟悉 HDFS 的常用操作（参考 Hdoop 实战 第 31-36 页），运行 WordCount 程序，得到统计结果。
请详细写出你每一步的操作，最好有截图，最后的结果部分必须有截图。

题目二：
实现一个统计输入文件中各个长度的单词出现频次的程序，实现一个生成随机字符串的程序

只需要在原有的 WordCount.java 文件进行少量修改，统计每个字符串对应的长度，并输出到屏幕

自行实现一个生成随机字符串的 Python 程序

二、核心代码：

题目一:
WordCount.java 文件，无需修改

题目二：
WordCount1.java 文件的核心部分

```
 public static class TokenizerMapper
      extends Mapper<Object, Text, Text, IntWritable>{

   private final static IntWritable one = new IntWritable(1);
   private Text word_length = new Text();

   public void map(Object key, Text value, Context context
                   ) throws IOException, InterruptedException {
     StringTokenizer itr = new StringTokenizer(value.toString());
     while (itr.hasMoreTokens()) {
       // word.set(itr.nextToken());
```

```
        // context.write(word, one);
        word_length.set(Integer.toString(itr.nextToken().length()));
        context.write(word_length, one);
      }
    }
  }
```

生成随机字符串的 Python 程序

```
import random
import string

with open("input.txt", "w") as f:
    for i in range (0, 10):
        print("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',random
.randint(1,10))))
        f.write("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',rand
om.randint(1,10))) + " ")
```

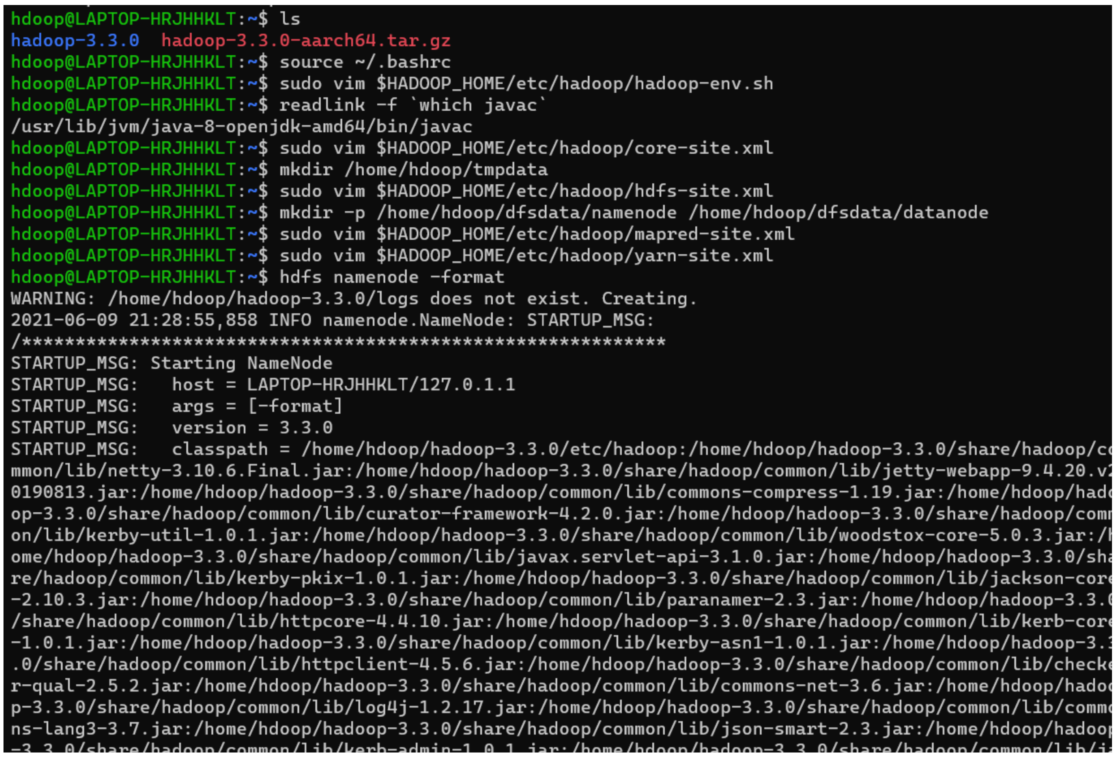
三、结果与分析:

题目一:

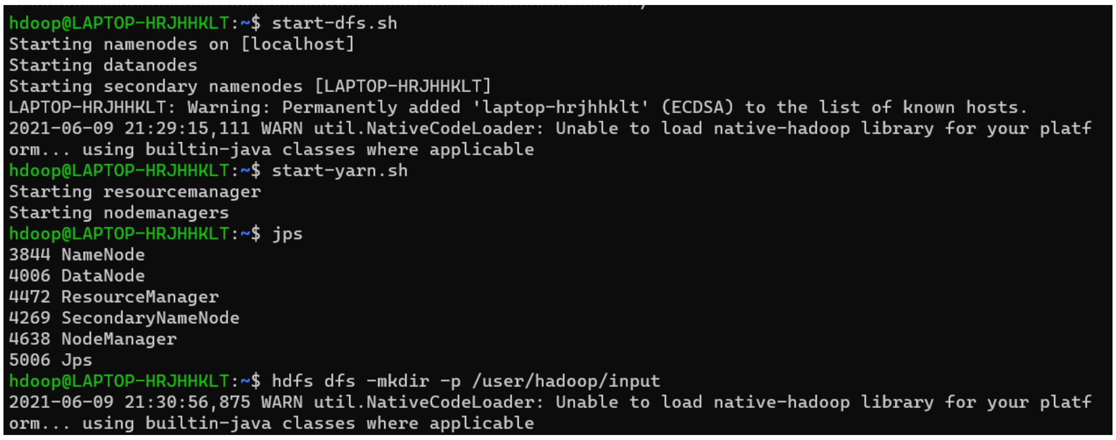
按照夏寒同学在群里发的 Hadoop 在 WSL2 上的配置博客一步一步进行配置

https://waltersumbon.github.io/2021/05/31/Hadoop%E5%AE%89%E8%A3%85%E6%8C%87%E5%8D%97/#more

这里给出部分中间过程的截图以及最后结果的截图

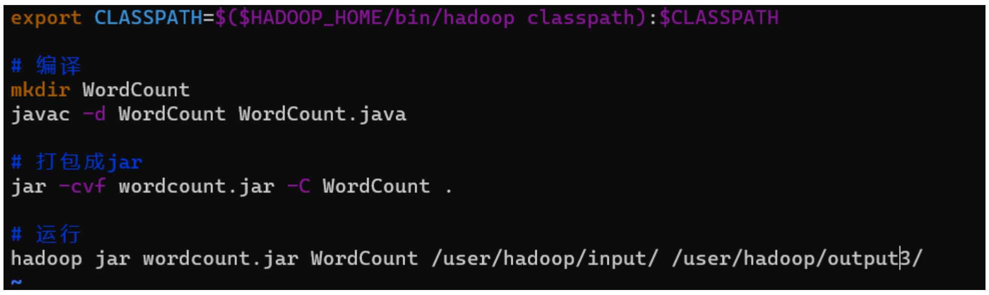


```
hdoop@LAPTOP-HRJHHKLT:~$ ls
hadoop-3.3.0  hadoop-3.3.0-aarch64.tar.gz
hdoop@LAPTOP-HRJHHKLT:~$ source ~/.bashrc
hdoop@LAPTOP-HRJHHKLT:~$ sudo vim $HADOOP_HOME/etc/hadoop/hadoop-env.sh
hdoop@LAPTOP-HRJHHKLT:~$ readlink -f `which javac`
/usr/lib/jvm/java-8-openjdk-amd64/bin/javac
hdoop@LAPTOP-HRJHHKLT:~$ sudo vim $HADOOP_HOME/etc/hadoop/core-site.xml
hdoop@LAPTOP-HRJHHKLT:~$ mkdir /home/hdoop/tmpdata
hdoop@LAPTOP-HRJHHKLT:~$ sudo vim $HADOOP_HOME/etc/hadoop/hdfs-site.xml
hdoop@LAPTOP-HRJHHKLT:~$ mkdir -p /home/hdoop/dfsdata/namenode /home/hdoop/dfsdata/datanode
hdoop@LAPTOP-HRJHHKLT:~$ sudo vim $HADOOP_HOME/etc/hadoop/mapred-site.xml
hdoop@LAPTOP-HRJHHKLT:~$ sudo vim $HADOOP_HOME/etc/hadoop/yarn-site.xml
hdoop@LAPTOP-HRJHHKLT:~$ hdfs namenode -format
WARNING: /home/hdoop/hadoop-3.3.0/logs does not exist. Creating.
2021-06-09 21:28:55,858 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = LAPTOP-HRJHHKLT/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 3.3.0
STARTUP_MSG:   classpath = /home/hdoop/hadoop-3.3.0/etc/hadoop:/home/hdoop/hadoop-3.3.0/share/hadoop/co
mmon/lib/netty-3.10.6.Final.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/jetty-webapp-9.4.20.v2
0190813.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/commons-compress-1.19.jar:/home/hdoop/hado
op-3.3.0/share/hadoop/common/lib/curator-framework-4.2.0.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/comm
on/lib/kerby-util-1.0.1.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/woodstox-core-5.0.3.jar:/h
ome/hdoop/hadoop-3.3.0/share/hadoop/common/lib/javax.servlet-api-3.1.0.jar:/home/hdoop/hadoop-3.3.0/sha
re/hadoop/common/lib/kerby-pkix-1.0.1.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/jackson-core
-2.10.3.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/paranamer-2.3.jar:/home/hdoop/hadoop-3.3.0
/share/hadoop/common/lib/httpcore-4.4.10.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/kerb-core
-1.0.1.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/kerby-asn1-1.0.1.jar:/home/hdoop/hadoop-3.3
.0/share/hadoop/common/lib/httpclient-4.5.6.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/checke
r-qual-2.5.2.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/commons-net-3.6.jar:/home/hdoop/hado
p-3.3.0/share/hadoop/common/lib/log4j-1.2.17.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/commo
ns-lang3-3.7.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/json-smart-2.3.jar:/home/hdoop/hadoop
-3.3.0/share/hadoop/common/lib/kerb-admin-1.0.1.jar:/home/hdoop/hadoop-3.3.0/share/hadoop/common/lib/ja
```



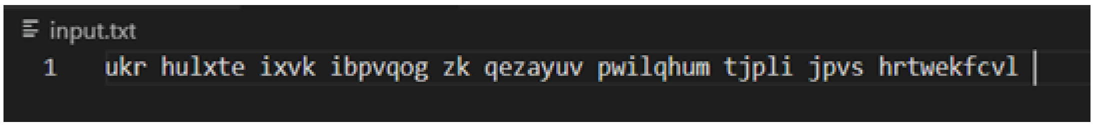
```
hdoop@LAPTOP-HRJHHKLT:~$ start-dfs.sh
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [LAPTOP-HRJHHKLT]
LAPTOP-HRJHHKLT: Warning: Permanently added 'laptop-hrjhhklt' (ECDSA) to the list of known hosts.
2021-06-09 21:29:15,111 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platf
orm... using builtin-java classes where applicable
hdoop@LAPTOP-HRJHHKLT:~$ start-yarn.sh
Starting resourcemanager
Starting nodemanagers
hdoop@LAPTOP-HRJHHKLT:~$ jps
3844 NameNode
4006 DataNode
4472 ResourceManager
4269 SecondaryNameNode
4638 NodeManager
5006 Jps
hdoop@LAPTOP-HRJHHKLT:~$ hdfs dfs -mkdir -p /user/hadoop/input
2021-06-09 21:30:56,875 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platf
orm... using builtin-java classes where applicable
```

将运行过程打包成 shell 脚本 run.sh 并运行



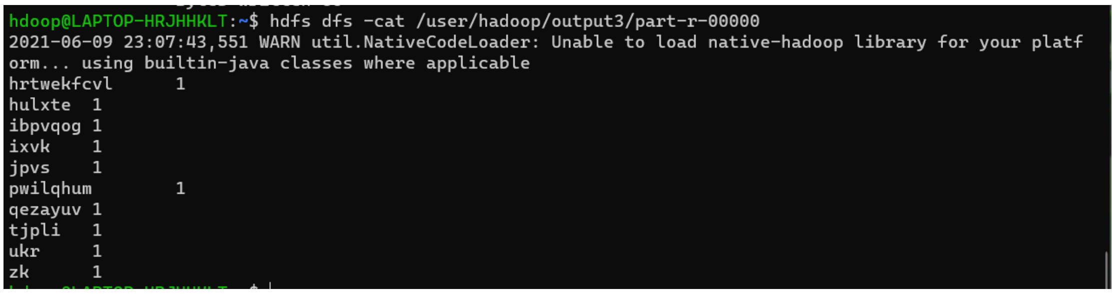
```
export CLASSPATH=$($HADOOP_HOME/bin/hadoop classpath):$CLASSPATH

# 编译
mkdir WordCount
javac -d WordCount WordCount.java

# 打包成jar
jar -cvf wordcount.jar -C WordCount .

# 运行
hadoop jar wordcount.jar WordCount /user/hadoop/input/ /user/hadoop/output3/
~
```

随机生成的输入文件如下



```
input.txt
ukr hulxte ixvk ibpvqog zk qezayuv pwilqhum tjpli jpvs hrtwekfcvl
```

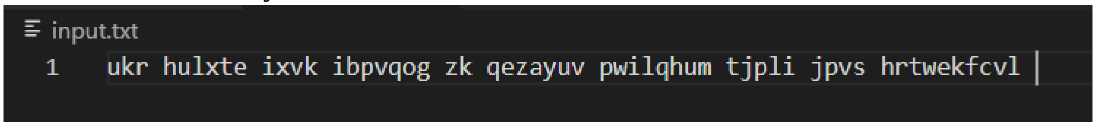
最终结果



```
hdoop@LAPTOP-HRJHHKLT:~$ hdfs dfs -cat /user/hadoop/output3/part-r-00000
2021-06-09 23:07:43,551 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platf
orm... using builtin-java classes where applicable
hrtwekfcvl      1
hulxte  1
ibpvqog 1
ixvk    1
jpvs    1
pwilqhum        1
qezayuv 1
tjpli   1
ukr     1
zk      1
```

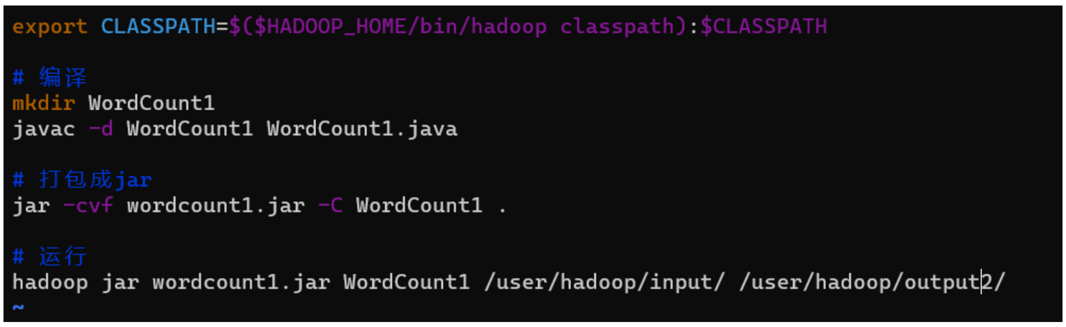
题目二：

利用上面给出的 Python 程序生成随机的输入文件



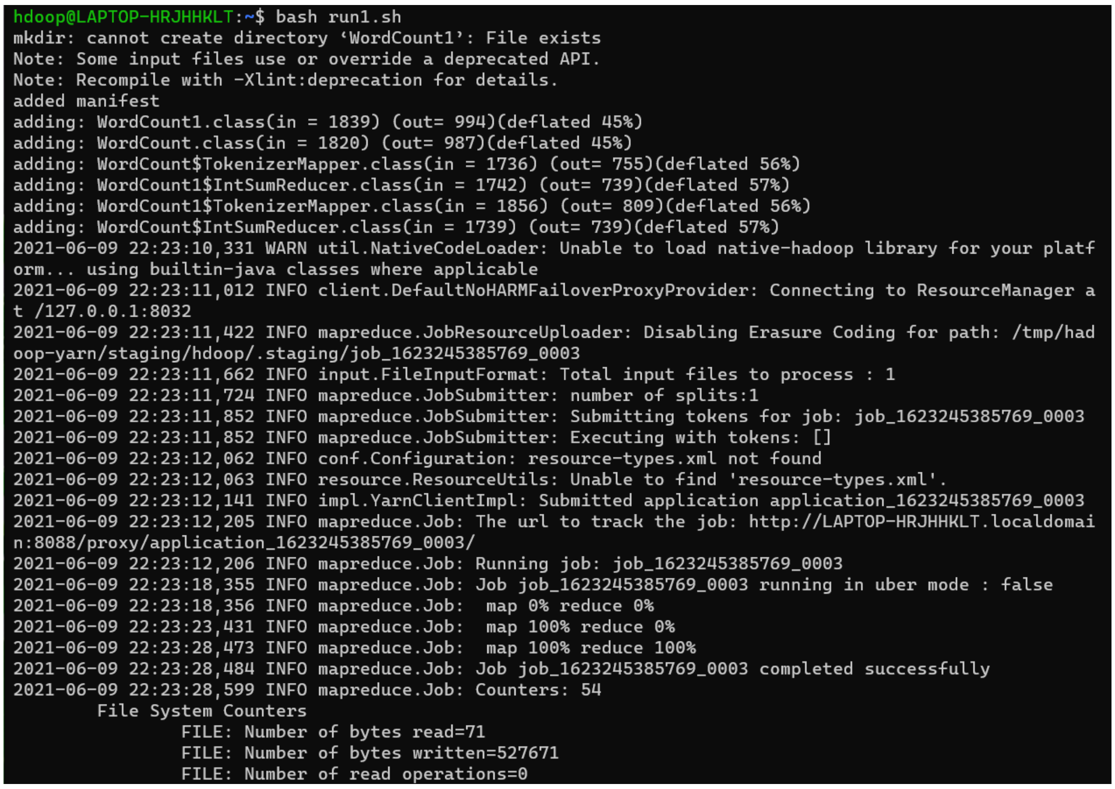
```
input.txt
ukr hulxte ixvk ibpvqog zk qezayuv pwilqhum tjpli jpvs hrtwekfcvl
```

将修改后的 Java 程序命名为 WordCount1.java，并将运行过程打包成 shell 脚本 run1.sh:



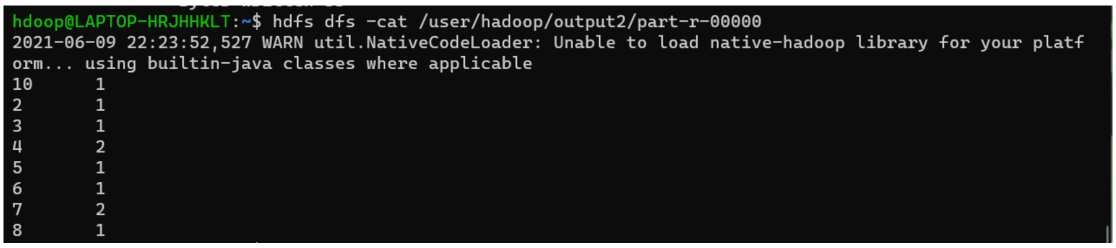
```
export CLASSPATH=$($HADOOP_HOME/bin/hadoop classpath):$CLASSPATH

# 编译
mkdir WordCount1
javac -d WordCount1 WordCount1.java

# 打包成jar
jar -cvf wordcount1.jar -C WordCount1 .

# 运行
hadoop jar wordcount1.jar WordCount1 /user/hadoop/input/ /user/hadoop/output2/
~
```

运行 run1.sh



```
hdoop@LAPTOP-HRJHHKLT:~$ bash run1.sh
mkdir: cannot create directory ‘WordCount1’: File exists
Note: Some input files use or override a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
added manifest
adding: WordCount1.class(in = 1839) (out= 994)(deflated 45%)
adding: WordCount.class(in = 1820) (out= 987)(deflated 45%)
adding: WordCount$TokenizerMapper.class(in = 1736) (out= 755)(deflated 56%)
adding: WordCount1$IntSumReducer.class(in = 1742) (out= 739)(deflated 57%)
adding: WordCount1$TokenizerMapper.class(in = 1856) (out= 809)(deflated 56%)
adding: WordCount$IntSumReducer.class(in = 1739) (out= 739)(deflated 57%)
2021-06-09 22:23:10,331 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platf
orm... using builtin-java classes where applicable
2021-06-09 22:23:11,012 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager a
t /127.0.0.1:8032
2021-06-09 22:23:11,422 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/had
oop-yarn/staging/hdoop/.staging/job_1623245385769_0003
2021-06-09 22:23:11,662 INFO input.FileInputFormat: Total input files to process : 1
2021-06-09 22:23:11,724 INFO mapreduce.JobSubmitter: number of splits:1
2021-06-09 22:23:11,852 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1623245385769_0003
2021-06-09 22:23:11,852 INFO mapreduce.JobSubmitter: Executing with tokens: []
2021-06-09 22:23:12,062 INFO conf.Configuration: resource-types.xml not found
2021-06-09 22:23:12,063 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2021-06-09 22:23:12,141 INFO impl.YarnClientImpl: Submitted application application_1623245385769_0003
2021-06-09 22:23:12,205 INFO mapreduce.Job: The url to track the job: http://LAPTOP-HRJHHKLT.localdomai
n:8088/proxy/application_1623245385769_0003/
2021-06-09 22:23:12,206 INFO mapreduce.Job: Running job: job_1623245385769_0003
2021-06-09 22:23:18,355 INFO mapreduce.Job: Job job_1623245385769_0003 running in uber mode : false
2021-06-09 22:23:18,356 INFO mapreduce.Job:  map 0% reduce 0%
2021-06-09 22:23:23,431 INFO mapreduce.Job:  map 100% reduce 0%
2021-06-09 22:23:28,473 INFO mapreduce.Job:  map 100% reduce 100%
2021-06-09 22:23:28,484 INFO mapreduce.Job: Job job_1623245385769_0003 completed successfully
2021-06-09 22:23:28,599 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=71
                FILE: Number of bytes written=527671
                FILE: Number of read operations=0
```

结果



```
hdoop@LAPTOP-HRJHHKLT:~$ hdfs dfs -cat /user/hadoop/output2/part-r-00000
2021-06-09 22:23:52,527 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platf
orm... using builtin-java classes where applicable
10      1
2       1
3       1
4       2
5       1
6       1
7       2
8       1
```

四、备注（* 可选）：

有可能影响结论的因素：

无

总结：
通过本次实验，对 Hadoop 的使用有了一定的了解。配置过程较为繁琐。最终实现时只需要很少量的修改，需要对 Java 有一定的了解。整体来看，收获颇丰

附录（源代码）

算法源代码（C/C++/JAVA 描述）
统计输入文件中各个长度的单词出现频次的程序

```java
import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.apache.hadoop.util.GenericOptionsParser;

public class WordCount1 {

  public static class TokenizerMapper
       extends Mapper<Object, Text, Text, IntWritable>{

    private final static IntWritable one = new IntWritable(1);
    private Text word_length = new Text();

    public void map(Object key, Text value, Context context
                    ) throws IOException, InterruptedException {
      StringTokenizer itr = new StringTokenizer(value.toString());
      while (itr.hasMoreTokens()) {
        // word.set(itr.nextToken());
        // context.write(word, one);
        word_length.set(Integer.toString(itr.nextToken().le
ngth()));
        context.write(word_length, one);
      }
    }
  }

  public static class IntSumReducer
       extends Reducer<Text,IntWritable,Text,IntWritable> {
    private IntWritable result = new IntWritable();

    public void reduce(Text key, Iterable<IntWritable> valu
es,
                       Context context
                       ) throws IOException, InterruptedExc
eption {
      int sum = 0;
      for (IntWritable val : values) {
        sum += val.get();
      }
      result.set(sum);
      context.write(key, result);
    }
  }

  public static void main(String[] args) throws Exception {
    Configuration conf = new Configuration();
    String[] otherArgs = new GenericOptionsParser(conf, arg
s).getRemainingArgs();
    if (otherArgs.length != 2) {
      System.err.println("Usage: wordcount <in> <out>");
      System.exit(2);
    }
    Job job = new Job(conf, "word count");
    job.setJarByClass(WordCount.class);
    job.setMapperClass(TokenizerMapper.class);
    job.setCombinerClass(IntSumReducer.class);
    job.setReducerClass(IntSumReducer.class);
    job.setOutputKeyClass(Text.class);
    job.setOutputValueClass(IntWritable.class);
    FileInputFormat.addInputPath(job, new Path(otherArgs[0]
));
    FileOutputFormat.setOutputPath(job, new Path(otherArgs[
1]));
    System.exit(job.waitForCompletion(true) ? 0 : 1);
  }
}
```

生成随机字符串的 Python 代码

```
import random
import string

with open("input.txt", "w") as f:
    for i in range (0, 10):
        print("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',random.randint(1,10))))
        f.write("".join(random.sample('zyxwvutsrqponmlkjihgfedcba',random.randint(1,10))) + " ")
```