Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
УРАЛЬСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
ИМЕНИ ПЕРВОГО ПРЕЗИДЕНТА РОССИИ Б.Н. ЕЛЬЦИНА
(УрФУ имени первого Президента России Б.Н. Ельцина)
Институт радиоэлектроники и информационных технологий — РТФ

ОТЧЁТ

по лабораторной работе №3

по дисциплине «Методы и инструменты анализа больших данных»

Преподаватель _____ _____ С.Г. Мирвода
(дата) (подпись)

Студент _____ _____ А.М. Белоусов
(дата) (подпись)

Студент _____ _____ А.В. Жиденко
(дата) (подпись)

Группа: РИМ-201211

Екатеринбург 2021

Цель работы: знакомство MapReduce.

Задание 0

Проверка полигона

1. Открыть презентацию по HIVE и повторить команды со слайдов про Mapreduce\YARN

export YARN_EXAMPLES=${HADOOP_HOME}/share/hadoop/mapreduce

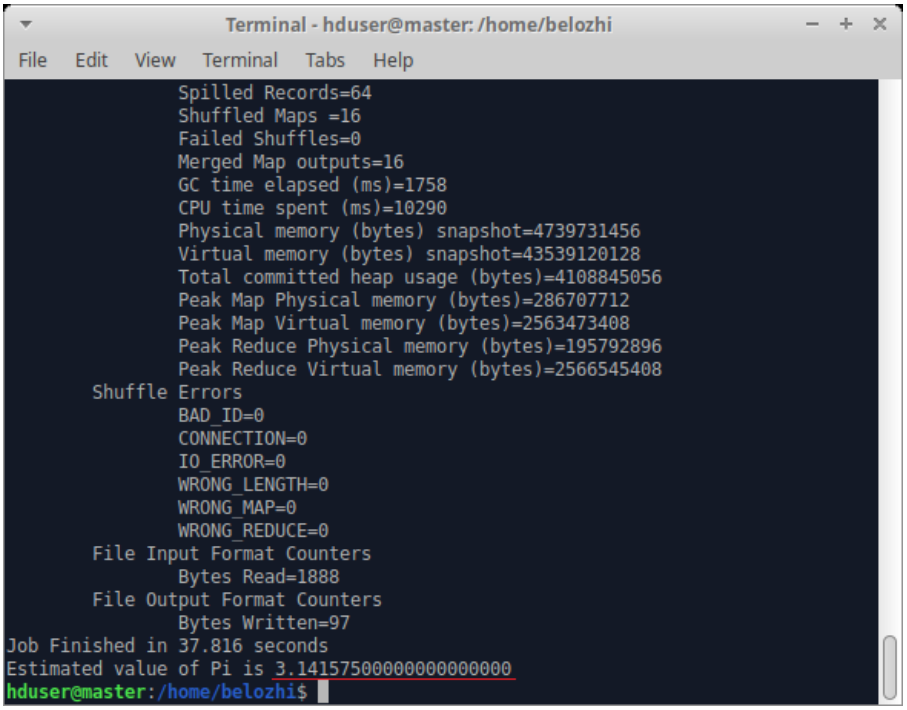
yarn jar ${YARN_EXAMPLES}/hadoop-mapreduce-examples-3.3.0.jar

```
hduser@master:/home/belozhi$ export YARN_EXAMPLES=${HADOOP_HOME}/share/hadoop/mapreduce
hduser@master:/home/belozhi$ yarn jar ${YARN_EXAMPLES}/hadoop-mapreduce-examples-3.3.0.jar
An example program must be given as the first argument.
Valid program names are:
  aggregatewordcount: An Aggregate based map/reduce program that counts the words in the input files.
  aggregatewordhist: An Aggregate based map/reduce program that computes the histogram of the words in the input files.
  bbp: A map/reduce program that uses Bailey-Borwein-Plouffe to compute exact digits of Pi.
  dbcount: An example job that count the pageview counts from a database.
  distbbp: A map/reduce program that uses a BBP-type formula to compute exact bits of Pi.
  grep: A map/reduce program that counts the matches of a regex in the input.
  join: A job that effects a join over sorted, equally partitioned datasets
  multifilewc: A job that counts words from several files.
  pentomino: A map/reduce tile laying program to find solutions to pentomino problems.
  pi: A map/reduce program that estimates Pi using a quasi-Monte Carlo method.
  randomtextwriter: A map/reduce program that writes 10GB of random textual data per node.
  randomwriter: A map/reduce program that writes 10GB of random data per node.
  secondarysort: An example defining a secondary sort to the reduce.
  sort: A map/reduce program that sorts the data written by the random writer.
  sudoku: A sudoku solver.
  teragen: Generate data for the terasort
  terasort: Run the terasort
  teravalidate: Checking results of terasort
  wordcount: A map/reduce program that counts the words in the input files.
  wordmean: A map/reduce program that counts the average length of the words in the input files.
  wordmedian: A map/reduce program that counts the median length of the words in the input files.
  wordstandarddeviation: A map/reduce program that counts the standard deviation of the length of the words in the input files.
```

yarn jar ${YARN_EXAMPLES}/hadoop-mapreduce-examples-3.3.0.jar pi 16 10000

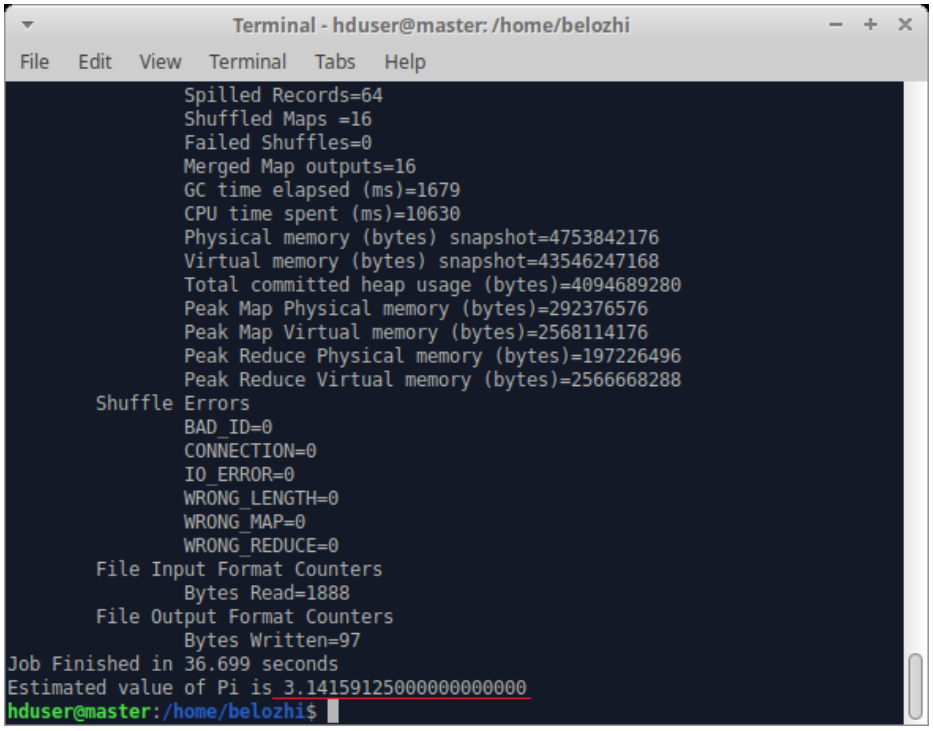
```
Terminal - hduser@master: /home/belozhi
File  Edit  View  Terminal  Tabs  Help
                Shuffled Maps =16
                Failed Shuffles=0
                Merged Map outputs=16
                GC time elapsed (ms)=2516
                CPU time spent (ms)=12350
                Physical memory (bytes) snapshot=4738502656
                Virtual memory (bytes) snapshot=43545952256
                Total committed heap usage (bytes)=4202168320
                Peak Map Physical memory (bytes)=288034816
                Peak Map Virtual memory (bytes)=2561871872
                Peak Reduce Physical memory (bytes)=191119360
                Peak Reduce Virtual memory (bytes)=2566533120
        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=1888
        File Output Format Counters
                Bytes Written=97
Job Finished in 48.415 seconds
Estimated value of Pi is 3.14127500000000000000
hduser@master:/home/belozhi$
hduser@master:/home/belozhi$
```

yarn jar ${YARN_EXAMPLES}/hadoop-mapreduce-examples-3.3.0.jar pi 16 100000



2. Выполнить пример расчёта pi с числом сэмплов 1M.

yarn jar ${YARN_EXAMPLES}/hadoop-mapreduce-examples-3.3.0.jar pi 16 1000000



Вычисленное значение Pi: 3.14159125000000000000

Задание 1

Выполнить MR задачу с записью данных в HDFS

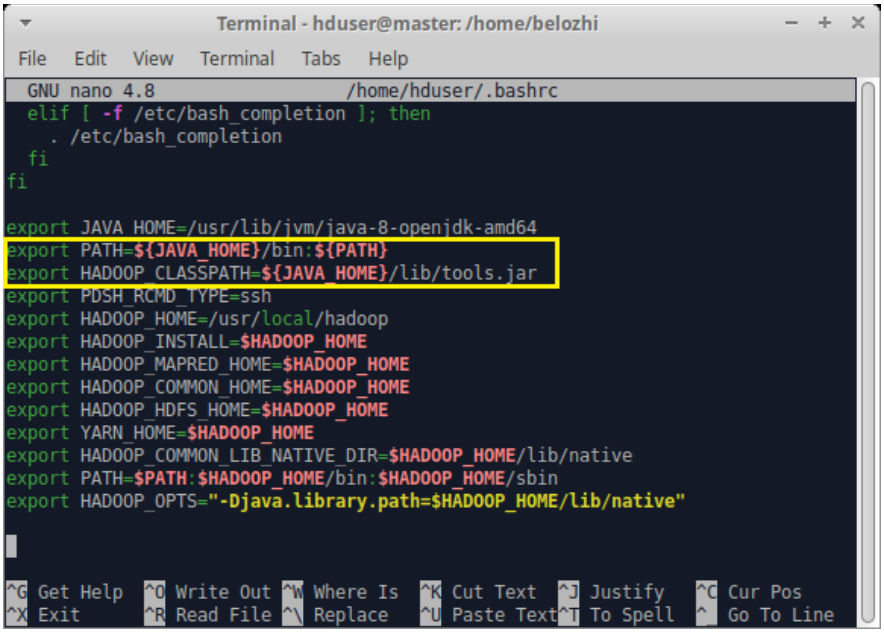
1. Прочитайте описание примера wordcount

Выполним настройку согласно примеру.

Добавим строки в файл */.bashrc* на master, node1, node2:

export PATH=${JAVA_HOME}/bin:${PATH}

export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar



На Master создаем файл /usr/local/hadoop/WordCount.java и копируем в него Source Code из примера wordcount.

```
Terminal - hduser@master: /home/belozhi
File  Edit  View  Terminal  Tabs  Help
  GNU nano 4.8              /usr/local/hadoop/WordCount.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;

public class WordCount {

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

    private final static IntWritable one = new IntWritable(1);
    private Text word = 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);
      }
    }
                                  [ Read 61 lines ]
^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell   ^_ Go To Line
```

2. Подготовьте тестовые папки в HDFS для запуска задачи и положите в папку input любой текстовый файл для анализа

```
belozhi (только для чтения) - LibreOffice Writer
Файл  Правка  Вид  Вставка  Формат  Стили  Таблица  Форма  Сервис  Окно  Справка

Anomalies are patterns in data that do not follow the expected behaviour and
they are rarely encountered. Anomaly detection has been widely used within
diverse research areas such as credit card fraud detection, image processing,
and many other application domains. In this paper, we focus on detecting
anomalies in power consumption data. The identification of unusual behaviours is
important in order to foresee uncommon events and to improve energy efficiency.
To this end, we propose a model to precisely identify anomalous days and another
one to localize the detected anomalies. Normal days are identified using a
simple Auto-Encoder reconstruction technique, whereas the localization of the
anomaly throughout the day is performed using a combination of LSTM and K-means
algorithms.

Страница 1 из 1   119 слов, 789 символов   Базовый   100 %
```

2.1. hadoop fs -mkdir -p /user/hadoop/wordcount/input

```
hduser@master:/home/belozhi$ hadoop fs -mkdir -p /user/hadoop/wordcount/input
```

2.2. hadoop fs -mkdir -p /user/hadoop/wordcount/output

```
hduser@master:/home/belozhi$ hadoop fs -mkdir -p /user/hadoop/wordcount/output
```

Папку output в дальнейшем пришлось удалить, т.к. Hadoop ругался на её существование.

```
hduser@master:/usr/local/hadoop$ hadoop jar wc.jar WordCount /user/hadoop/wordcount/input /user/hadoop/wordcount/output/
2021-12-18 18:41:51,969 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at master/192.168.121.16:8032
Exception in thread "main" org.apache.hadoop.mapred.FileAlreadyExistsException: Output directory hdfs://master:9000/user/hadoop/wordcount/output
already exists
        at org.apache.hadoop.mapreduce.lib.output.FileOutputFormat.checkOutputSpecs(FileOutputFormat.java:164)
        at org.apache.hadoop.mapreduce.JobSubmitter.checkSpecs(JobSubmitter.java:277)
        at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:143)
        at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1576)
        at org.apache.hadoop.mapreduce.Job$11.run(Job.java:1573)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:422)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1845)
        at org.apache.hadoop.mapreduce.Job.submit(Job.java:1573)
        at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1594)
        at WordCount.main(WordCount.java:59)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:323)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
```

2.3. hadoop fs -put ВАШ_ФАЙЛ /user/hadoop/wordcount/input

```
hduser@node1:~$ hadoop fs -put belozhi /user/hadoop/wordcount/input
```

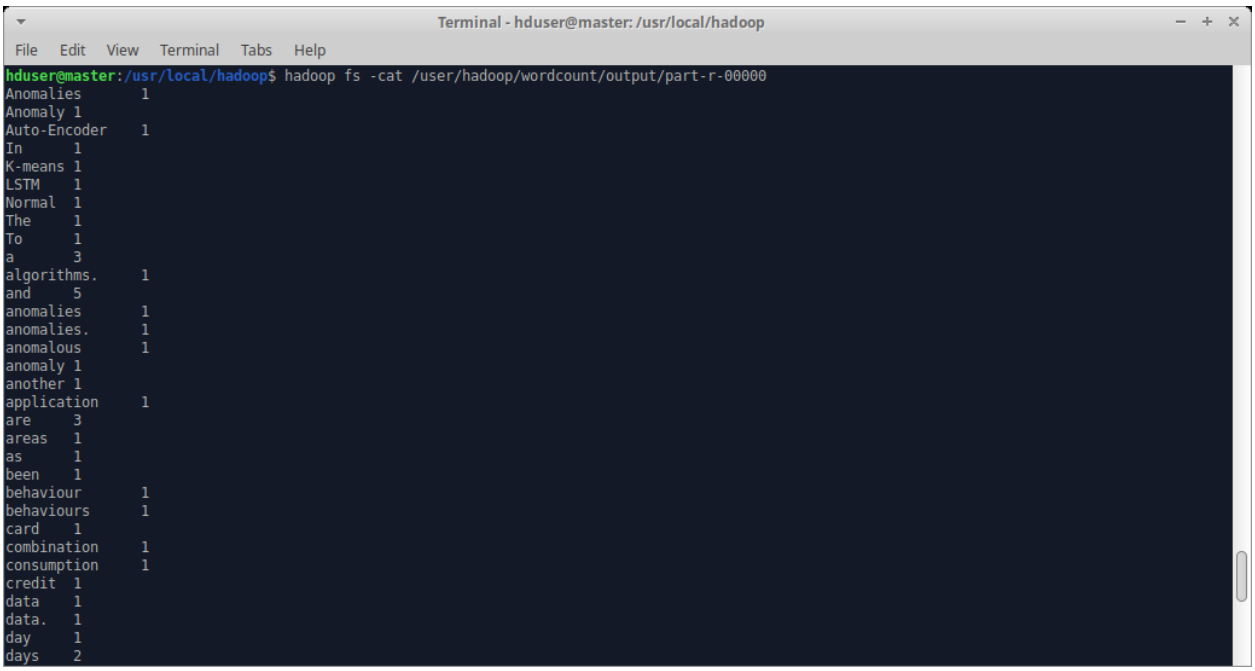
3. Запустите пример wordcount по аналогии с примером выше.

```
Terminal - hduser@master: /usr/local/hadoop
File  Edit  View  Terminal  Tabs  Help
        at java.lang.reflect.Method.invoke(Method.java:498)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:323)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:236)
hduser@master:/usr/local/hadoop$ hadoop jar wc.jar WordCount /user/hadoop/wordcount/input /user/hadoop/wordcount/output/
2021-12-18 18:42:25,036 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at master/192.168.121.16:8032
2021-12-18 18:42:25,499 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and ex
ecute your application with ToolRunner to remedy this.
2021-12-18 18:42:25,525 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/hduser/.staging/job_16398
32184604_0001
2021-12-18 18:42:25,905 INFO input.FileInputFormat: Total input files to process : 1
2021-12-18 18:42:26,019 INFO mapreduce.JobSubmitter: number of splits:1
2021-12-18 18:42:26,263 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1639832184604_0001
2021-12-18 18:42:26,263 INFO mapreduce.JobSubmitter: Executing with tokens: []
2021-12-18 18:42:26,526 INFO conf.Configuration: resource-types.xml not found
2021-12-18 18:42:26,527 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2021-12-18 18:42:26,892 INFO impl.YarnClientImpl: Submitted application application_1639832184604_0001
2021-12-18 18:42:26,973 INFO mapreduce.Job: The url to track the job: http://master:8088/proxy/application_1639832184604_0001/
2021-12-18 18:42:26,973 INFO mapreduce.Job: Running job: job_1639832184604_0001
2021-12-18 18:42:36,190 INFO mapreduce.Job: Job job_1639832184604_0001 running in uber mode : false
2021-12-18 18:42:36,195 INFO mapreduce.Job:  map 0% reduce 0%
2021-12-18 18:42:44,322 INFO mapreduce.Job:  map 100% reduce 0%
2021-12-18 18:42:50,393 INFO mapreduce.Job:  map 100% reduce 100%
2021-12-18 18:42:51,429 INFO mapreduce.Job: Job job_1639832184604_0001 completed successfully
2021-12-18 18:42:51,558 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=1279
                FILE: Number of bytes written=530431
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=909
                HDFS: Number of bytes written=893
                HDFS: Number of read operations=8
```

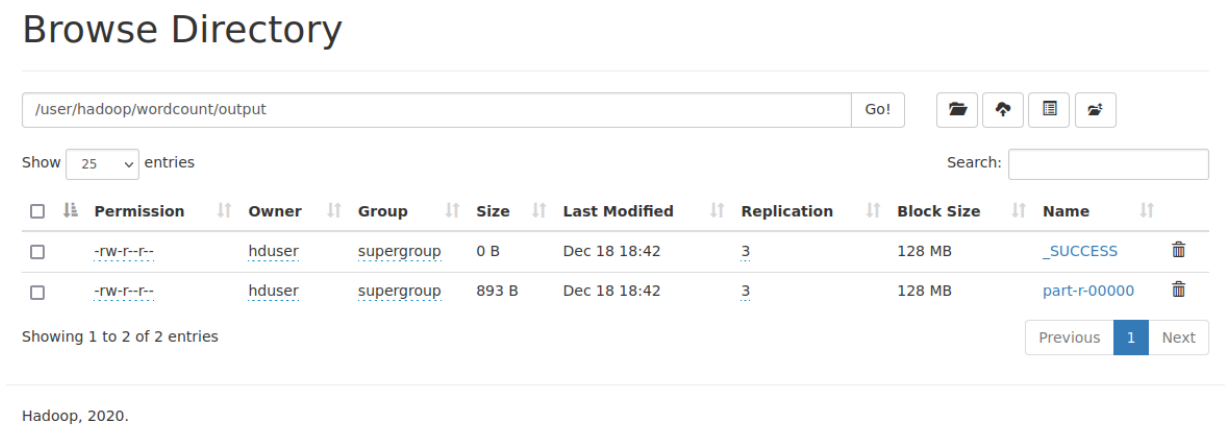
```
Terminal - hduser@master: /usr/local/hadoop
File  Edit  View  Terminal  Tabs  Help
                HDFS: Number of read operations=8
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=1
                Launched reduce tasks=1
                Data-local map tasks=1
                Total time spent by all maps in occupied slots (ms)=5377
                Total time spent by all reduces in occupied slots (ms)=3061
                Total time spent by all map tasks (ms)=5377
                Total time spent by all reduce tasks (ms)=3061
                Total vcore-milliseconds taken by all map tasks=5377
                Total vcore-milliseconds taken by all reduce tasks=3061
                Total megabyte-milliseconds taken by all map tasks=5506048
                Total megabyte-milliseconds taken by all reduce tasks=3134464
        Map-Reduce Framework
                Map input records=1
                Map output records=119
                Map output bytes=1266
                Map output materialized bytes=1279
                Input split bytes=119
                Combine input records=119
                Combine output records=95
                Reduce input groups=95
                Reduce shuffle bytes=1279
                Reduce input records=95
                Reduce output records=95
                Spilled Records=190
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=276
```

```
Terminal - hduser@master: /usr/local/hadoop
File  Edit  View  Terminal  Tabs  Help
                Map output materialized bytes=1279
                Input split bytes=119
                Combine input records=119
                Combine output records=95
                Reduce input groups=95
                Reduce shuffle bytes=1279
                Reduce input records=95
                Reduce output records=95
                Spilled Records=190
                Shuffled Maps =1
                Failed Shuffles=0
                Merged Map outputs=1
                GC time elapsed (ms)=276
                CPU time spent (ms)=1930
                Physical memory (bytes) snapshot=467992576
                Virtual memory (bytes) snapshot=5128511488
                Total committed heap usage (bytes)=407371776
                Peak Map Physical memory (bytes)=283525120
                Peak Map Virtual memory (bytes)=2560704512
                Peak Reduce Physical memory (bytes)=184467456
                Peak Reduce Virtual memory (bytes)=2567806976
        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=790
        File Output Format Counters
                Bytes Written=893
hduser@master:/usr/local/hadoop$ hadoop fs -cat /user/hadoop/wordcount/output/part-r-00000
```

4. После завершения просмотрите результаты в папке output и в отчёт включите несколько первых строк из файла результата (обычно называется part-r-00000)



```
Terminal - hduser@master: /usr/local/hadoop
File  Edit  View  Terminal  Tabs  Help
hduser@master:/usr/local/hadoop$ hadoop fs -cat /user/hadoop/wordcount/output/part-r-00000
Anomalies       1
Anomaly 1
Auto-Encoder    1
In      1
K-means 1
LSTM    1
Normal  1
The     1
To      1
a       3
algorithms.     1
and     5
anomalies       1
anomalies.      1
anomalous       1
anomaly 1
another 1
application     1
are     3
areas   1
as      1
been    1
behaviour       1
behaviours      1
card    1
combination     1
consumption     1
credit  1
data    1
data.   1
day     1
days    2
```



Browse Directory

/user/hadoop/wordcount/output Go!

Show 25 entries Search:

	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
☐	-rw-r--r--	hduser	supergroup	0 B	Dec 18 18:42	3	128 MB	_SUCCESS
☐	-rw-r--r--	hduser	supergroup	893 B	Dec 18 18:42	3	128 MB	part-r-00000

Showing 1 to 2 of 2 entries Previous 1 Next

Hadoop, 2020.