

به نام ایزد بخشاينده مهربان



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

گزارش تمرین دوم

درس مبانی رایانش ابری

استاد: دکتر سید احمد جوادی

اعضای گروه:

علی شفیعی - ۹۵۳۱۸۰۱

سید سجاد پیشوائیان - ۹۵۳۱۰۱۵

دی ۱۳۹۹

لازم به ذکر است که تمامی بخش‌ها با همکاری هر دو عضو با استفاده از skpye زده شده است و پیاده‌سازی کاملاً در هر بخش دو نفره و اشتراکی بوده است.

بخش اول:

سوال اول: از آنجا که اجرای عملیات روی I/O زمانبر است و Hadoop map-reduce نیز وابستگی زیادی به دیسک دارد، از طرف دیگر در اجرا به صورت map-reduce، ابتدا عملیات map باید کامل انجام شود و سپس عملیات reduce اجرا می‌شود که در صورتی که حجم داده‌ها بالا باشد، با توجه به اینکه map-reduce روی دیسک انجام می‌شود، بنابراین زمان قابل ملاحظه‌ای در این نوع اجرا تلف می‌شود.

سوال دوم: Apache Spark طبق گفته خود سازنده "یک موتور تجزیه و تحلیل متحد برای پردازش داده در مقیاس بزرگ" است. Spark توسط بنیاد غیرانتفاعی نرم افزار Apache که صدها پروژه نرم افزاری منبع باز منتشر کرده است، نگهداری می‌شود. بیش از ۱۲۰۰ توسعه دهنده از زمان شروع پروژه در Spark مشارکت داشته اند.

Spark که ابتدا در AMPLab UC Berkeley توسعه یافته بود، برای اولین بار در سال ۲۰۱۰ به عنوان یک پروژه منبع باز منتشر شد. Spark از چارچوب محاسبات توزیع شده Hadoop MapReduce به عنوان پایه و اساس خود استفاده می‌کند. Spark با هدف حفظ بسیاری از مزایای MapReduce برای بهبود جنبه های مختلف پروژه MapReduce مانند عملکرد و سهولت استفاده در نظر گرفته شد.

Spark شامل یک موتور پردازش اطلاعات اصلی و همچنین کتابخانه هایی برای SQL، یادگیری ماشین و پردازش جریان است. با استفاده از API برای جاوا، اسکالا، پایتون و R، Spark از جذابیت گسترده ای در بین توسعه دهندگان برخوردار است - به این ترتیب شهرت "چاقوی ارتش سوئیس" در پردازش داده های بزرگ را بدست آورد.

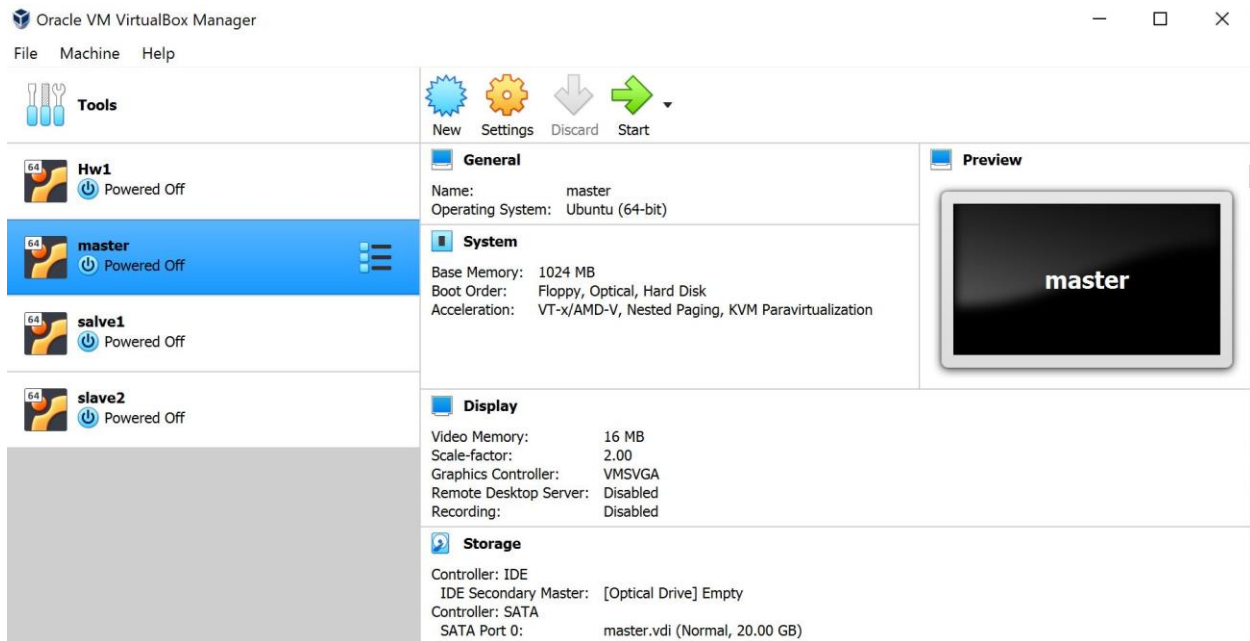
Apache Spark داده ها را در حافظه دسترسی تصادفی (RAM) پردازش می کند، در حالی که Hadoop MapReduce داده ها را پس از نقشه بر روی دیسک ادامه می دهد یا عملکرد را کاهش می دهد. بنابراین، از نظر تئوری، اسپارک از Hadoop MapReduce پیشی می گیرد.

با این وجود Spark به حافظه زیادی احتیاج دارد. Spark ، دقیقاً مانند پایگاه های داده استاندارد، فرایندی را در حافظه بارگذاری می کند و آن را تا اطلاع ثانوی به دلیل ذخیره سازی در آن نگه می دارد. اگر Spark در Hadoop YARN اجرا شود یا اگر داده ها خیلی بزرگ باشند و نتوانند کاملاً در حافظه جا بگیرند ، Spark ممکن است دچار افت شدید عملکرد شود. از طرف دیگر ، MapReduce به محض انجام کار فرایندهای خود را از بین می برد ، بنابراین می تواند به راحتی در کنار سایر سرویس ها با اختلاف عملکرد جزئی کار کند. Spark برای محاسبات تکراری که باید چندین بار از داده های مشابه عبور کند ، حرف اول را می زند. اما وقتی صحبت از مشاغل ETL یک مرحله ای - مثلاً تغییر شکل داده ها یا یکپارچه سازی داده ها - می شود، این دقیقاً همان چیزی است که MapReduce برای آن طراحی شده است. هنگامی که تمام داده ها در حافظه قرار می گیرند ، به ویژه در خوشه های اختصاصی ، Spark عملکرد بهتری دارد. Hadoop MapReduce برای داده هایی طراحی شده است که در حافظه جای نمی گیرند و می توانند در کنار سایر سرویس ها به خوبی اجرا شوند. مقایسه spark و Hadoop map-reduce:

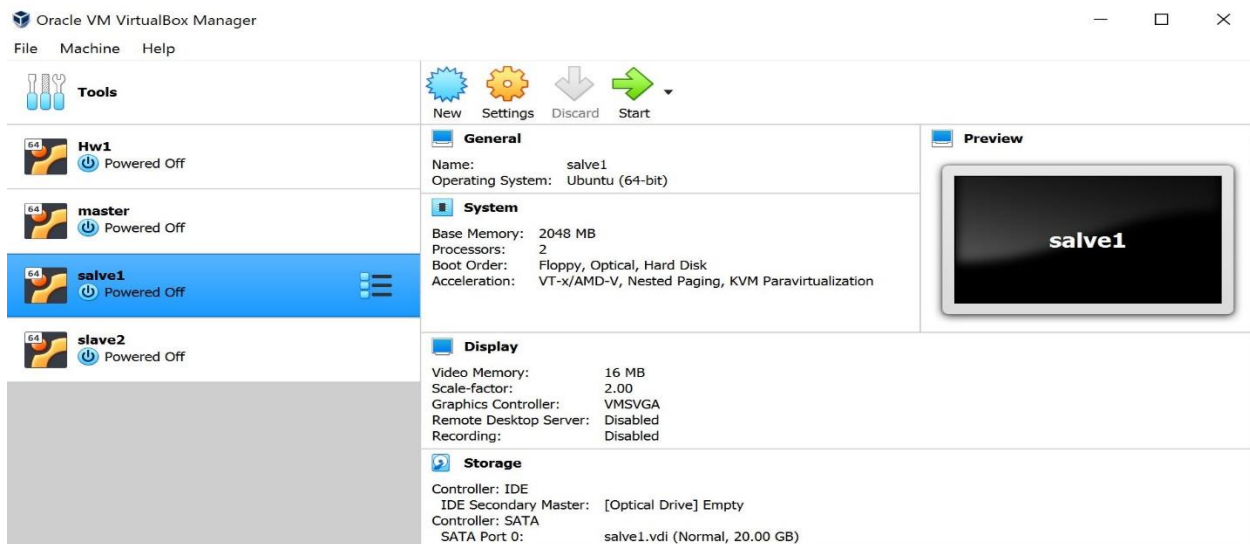
تجزیه و تحلیل real-time	Apache Spark می تواند داده های real-time را پردازش کند ، یعنی داده هایی که از جریان های رویداد در زمان واقعی می آیند ، با نرخ میلیون ها رویداد در ثانیه ، به عنوان مثال. داده های توییتر. قدرت Spark توانایی پردازش موثر جریانهای زنده است اما MapReduce با شکست مواجه می شود، زیرا برای انجام پردازش دسته ای بر روی حجم زیاد داده ها طراحی شده است.
تاخیر	Spark محاسبات کم تاخیر را فراهم می کند اما MapReduce یک چارچوب محاسباتی با تأخیر زیاد است.
استفاده	برنامه Spark آسان است زیرا دارای اپراتورهای سطح بالا با RDD - مجموعه داده های توزیع شده انعطاف پذیر (Resilient Distributed Dataset) است اما در MapReduce ، توسعه دهندگان باید هر عملیات را دستی کد کنند که کار در آن را بسیار دشوار می کند.
حالت تعاملی	Apache Spark می تواند داده ها را به صورت تعاملی پردازش کند اما MapReduce حالت تعاملی ندارد.

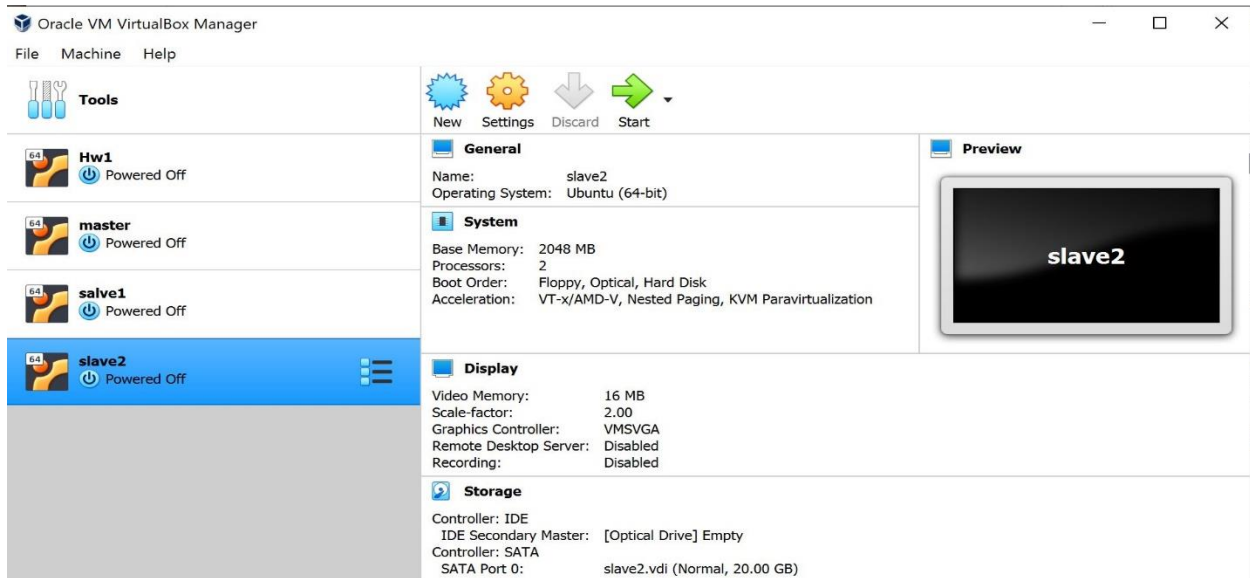
بخش دوم:

۱-۲- پیکربندی VM1(Master):



پیکربندی VM2(Slave1) و VM3(Slave2):





:VM1(Master) -۲-۲

```
h-user@master:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.20.10.3 netmask 255.255.255.240 broadcast 172.20.10.15
    inet6 fe80::a00:27ff:fe57:e733 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:57:e7:33 txqueuelen 1000 (Ethernet)
    RX packets 16833 bytes 24335338 (24.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5991 bytes 441862 (441.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 100 bytes 7812 (7.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 100 bytes 7812 (7.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

h-user@master:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as h-user in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [master]
Starting datanodes
Starting secondary namenodes [master]
Starting resourcemanager
Starting nodemanagers
h-user@master:~$ jps
1634 ResourceManager
1465 SecondaryNameNode
1211 NameNode
1917 Jps
h-user@master:~$
```

:VM3(Slave2) , VM2(Slave1)

```

h-user@slave1:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.20.10.5 netmask 255.255.255.240 broadcast 172.20.10.15
    inet6 fe80::a00:27ff:fe5:5249 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:f5:52:49 txqueuelen 1000 (Ethernet)
    RX packets 17 bytes 4054 (4.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 17 bytes 1882 (1.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 84 bytes 6324 (6.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 84 bytes 6324 (6.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

h-user@slave1:~$ jps
1041 DataNode
1299 Jps
1210 NodeManager
h-user@slave1:~$ _

```

```

ali@slave2:~$ su - h-user
Password:
h-user@slave2:~$ jps
2036 NodeManager
1866 DataNode
2125 Jps
h-user@slave2:~$

```

۳-۲- آدرس IP ماشین‌های مجازی ما به صورت زیر هستند:

Master	172.20.10.3
Slave1	172.20.10.5
Slave2	172.20.10.8

در اینجا HDFS و Hadoop NameNodes با از طریق پورت ۹۸۷۰ با مرورگر قابل دسترسی

هستند:

←→↻🏠

172.20.10.3:9870/dfshealth.html#tab-overview

⋮🔍🌟

⬇️🔍📄🔄🔌🔌🔌

⋮

Hadoop

Overview

Datanodes

Datanode Volume Failures

Snapshot

Startup Progress

Utilities

Overview

master:9000' (active)

Started:	Sat Dec 26 20:53:40 +0330 2020
Version:	3.2.1, rb3cbb467e22ea829b38084b7b01d07e0bf3842
Compiled:	Tue Sep 10 20:26:00 +0430 2019 by rohitsharmaks from branch-3.2.1
Cluster ID:	CID-d51b887d-b192-42ea-af48-ec7bd79cf905
Block Pool ID:	BP-971279446-172.20.10.3-1608994182002

Summary

Security is off.
Safemode is off.
8 files and directories, 2 blocks (2 replicated blocks, 0 erasure coded block groups) = 10 total filesystem object(s).
Heap Memory used 19.64 MB of 30.35 MB Heap Memory. Max Heap Memory is 237.81 MB.
Non Heap Memory used 47.68 MB of 49.06 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	37.15 GB
Configured Remote Capacity:	0 B
DFS Used:	112 KB (0%)
Non DFS Used:	11.71 GB
DFS Remaining:	23.5 GB (63.26%)
Block Pool Used:	112 KB (0%)
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	2 (Decommissioned: 0, In Maintenance: 0)
Dead Nodes	0 (Decommissioned: 0, In Maintenance: 0)

←→↻🏠

172.20.10.3:9870/dfshealth.html#tab-overview

⋮🔍🌟

⬇️🔍📄🔄🔌🔌🔌

⋮

Entering Maintenance Nodes

0

Total Datanode Volume Failures

0 (0 B)

Number of Under-Replicated Blocks

0

Number of Blocks Pending Deletion (including replicas)

0

Block Deletion Start Time

Sat Dec 26 20:53:40 +0330 2020

Last Checkpoint Time

Sat Dec 26 20:53:42 +0330 2020

Enabled Erasure Coding Policies

RS-6-3-1024k

NameNode Journal Status

Current transaction ID: 36

Journal Manager	State
FileJournalManager(root=/usr/local/hadoop/data/nameNode)	EditLogFileOutputStream(/usr/local/hadoop/data/nameNode/current/edits_inprogress_00000000000000000036)

NameNode Storage

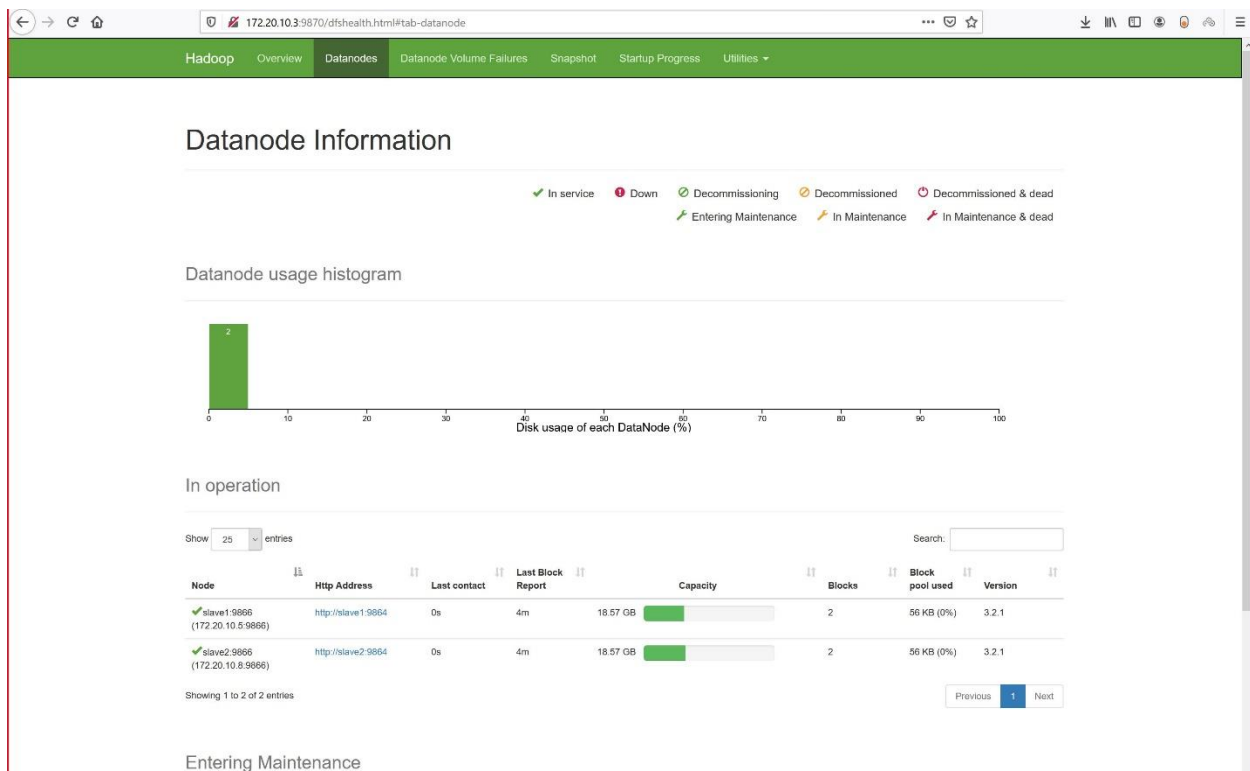
Storage Directory	Type	State
/usr/local/hadoop/data/nameNode	IMAGE_AND_EDITS	Active

DFS Storage Types

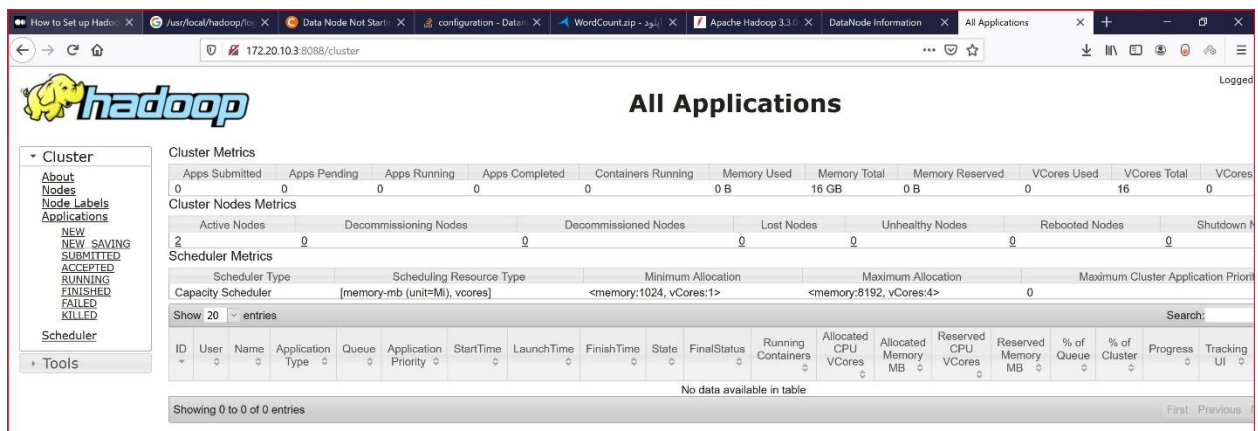
Storage Type	Configured Capacity	Capacity Used	Capacity Remaining	Block Pool Used	Nodes in Service
DISK	37.15 GB	112 KB (0%)	23.5 GB (63.26%)	112 KB	2

Hadoop, 2019.

DataNode نیز با همین آدرس و پورت ۹۸۶۴ از طریق مرورگر قابل دسترسی است:



ResourceManager نیز با آدرس قبل و پورت ۸۰۸۸ از طریق مرورگر قابل دسترسی است:



۴-۲- با دستور `Hadoop fs -mkdir /user/Hadoop` پوشه `/user/Hadoop` را در

HDFS ساختیم و فایل `test.txt` را ایجاد کرده و با دستور زیر در HDFS

`Hadoop fs -copyFromLocal /home/ali/test.txt /user/Hadoop/input`

```
h-user@master:~$ hadoop fs -copyFromLocal /home/ali/test.txt /user/hadoop/input
2020-12-26 15:27:40,038 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
h-user@master:~$ hadoop fs -ls -R /
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:07 /user
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:27 /user/hadoop
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:27 /user/hadoop/input
-rw-r--r-- 2 h-user supergroup        82 2020-12-26 15:27 /user/hadoop/input/test.txt
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:27 /user/hadoop/output
```

محتوای فایل `test.txt`:



```
ali@master:~$ cat test.txt
ali
sajjad
sajjad
ali
javadi
javadi
ali
sajjad
feat
ali
cloud
computing
fall
fall
ali@master:~$
```

فایل بارگذاری شده در HDFS:

```
h-user@master:~$ hadoop fs -cat /user/hadoop/input/test.txt
2020-12-26 17:46:25,813 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
ali
sajjad
sajjad
ali
javadi
javadi
ali
sajjad
feat
ali
cloud
computing
fall
fall
h-user@master:~$ _
```

فایل و پوشه ایجاد شده از طریق مرورگر نیز قابل مشاهده و دسترسی است:

The first screenshot shows the Hadoop web interface at the URL `172.20.103:9870/explorer.html#/user/hadoop`. The 'Browse Directory' section shows the path `/user/hadoop`. Below the search bar, there is a table with columns: Permission, Owner, Group, Size, Last Modified, Replication, Block Size, and Name. The table lists two entries: 'input' and 'output', both with permissions `drwxr-xr-x`, owner `h-user`, group `supergroup`, size `0 B`, and last modified `Dec 26 18:57` and `Dec 26 19:01` respectively. The second screenshot shows the same interface but at the URL `172.20.103:9870/explorer.html#/user/hadoop/input`. The table now shows a single entry: 'test.txt' with permissions `-rw-r--r--`, owner `h-user`, group `supergroup`, size `82 B`, last modified `Dec 26 18:57`, and replication `2`.

۵-۲- اجرای برنامه WordCount: فایل WordCount.java را با طبق لینک موجود در صورت تمرین در کامپیوتر شخصی ایجاد کردیم و آن را روی سایت <https://uupload.ir> آپلود کردیم و در Hadoop با دستور `wget` آن را دانلود و اکسپورت کردیم (لینک دانلود):

```
h-user@master:~$ wget https://upload.ir/filelink/FGgtzAVlqwnJ/iows_wordcount.zip
--2020-12-26 15:18:24-- https://upload.ir/filelink/FGgtzAVlqwnJ/iows_wordcount.zip
Resolving upload.ir (upload.ir)... 212.33.193.82
Connecting to upload.ir (upload.ir)|212.33.193.82|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 873 [application/octet-stream]
Saving to: 'iows_wordcount.zip.1'

iows_wordcount.zip.1  100%[=====]      873  --.-KB/s   in 0s

2020-12-26 15:18:24 (39.2 MB/s) - 'iows_wordcount.zip.1' saved [873/873]

h-user@master:~$ unzip iows_wordcount.zip.1
Archive:  iows_wordcount.zip.1
  inflating: WordCount.java
h-user@master:~$ ls
iows_wordcount.zip  iows_wordcount.zip.1  iow_wordcount.zip  WordCount.java
```

سپس دستورات زیر را قدم به قدم اجرا کردیم: (اسکرین)

```
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/
export PATH=${JAVA_HOME}/bin:${PATH}
export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar
```

با دستورات زیر نیز فایل jar برنامه را ایجاد کردیم: (اسکرین)

```
$ hadoop com.sun.tools.javac.Main WordCount.java
$ jar cf wc.jar WordCount*.class
```

```
h-user@master:~$ hadoop com.sun.tools.javac.Main WordCount.java
/usr/local/hadoop/bin/../libexec/hadoop-functions.sh: line 2366: HADOOP_COM.SUN.TOOLS.JAVAC.MAIN_USE
R: invalid variable name
/usr/local/hadoop/bin/../libexec/hadoop-functions.sh: line 2461: HADOOP_COM.SUN.TOOLS.JAVAC.MAIN_OPT
S: invalid variable name
h-user@master:~$ jar
```

با دستور `hadoop jar wc.jar WordCount /user/Hadoop/input`

`/user/Hadoop/output` نیز برنامه را اجرا کردیم:

بررسی صحت اجرا:

```
h-user@master:~$ hadoop fs -ls -R /
drwxr-xr-x - h-user supergroup      0 2020-12-26 15:07 /user
drwxr-xr-x - h-user supergroup      0 2020-12-26 15:31 /user/hadoop
drwxr-xr-x - h-user supergroup      0 2020-12-26 15:27 /user/hadoop/input
-rw-r--r-- 2 h-user supergroup    82 2020-12-26 15:27 /user/hadoop/input/test.txt
drwxr-xr-x - h-user supergroup      0 2020-12-26 15:31 /user/hadoop/output
-rw-r--r-- 2 h-user supergroup      0 2020-12-26 15:31 /user/hadoop/output/_SUCCESS
-rw-r--r-- 2 h-user supergroup    58 2020-12-26 15:31 /user/hadoop/output/part-r-00000
```

مشاهده می شود برنامه به درستی اجرا شده است.

خروجی برنامه:

```
h-user@master:~$ hadoop fs -cat /user/hadoop/output/part-r-00000
2020-12-26 17:45:19,061 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrus
ted = false, remoteHostTrusted = false
ali      4
cloud    1
computing      1
fall      2
feat      1
javadi     2
sajjad     3
```

:HDFS web gui

The screenshot shows the Hadoop web GUI interface. At the top, there's a navigation bar with links: Hadoop, Overview, Datanodes, Datanode Volume Failures, Snapshot, Startup Progress, and Utilities. The main content area is titled 'Browse Directory' and shows the path '/user/hadoop'. Below the path, there's a 'Go!' button and icons for file operations. A table lists the contents of the directory:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	h-user	supergroup	0 B	Dec 26 18:57	0	0 B	input
drwxr-xr-x	h-user	supergroup	0 B	Dec 26 19:01	0	0 B	output

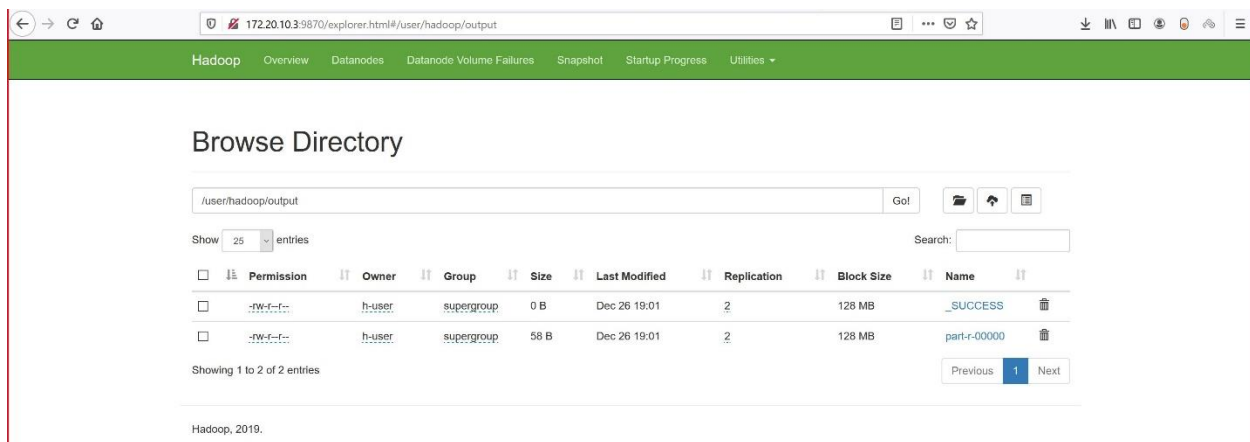
Below the table, it says 'Showing 1 to 2 of 2 entries'. There are 'Previous', '1', and 'Next' buttons. At the bottom, it says 'Hadoop, 2019.'.

Browse Directory

This screenshot shows the same Hadoop web GUI interface, but with a different file selected. The table now shows only one entry:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
-rw-r--r--	h-user	supergroup	82 B	Dec 26 18:41	2	128 MB	test.txt

Below the table, it says 'Showing 1 to 1 of 1 entries'. There are 'Previous', '1', and 'Next' buttons. At the bottom, it says 'Hadoop, 2019.'.



۲-۶- اجرای ضرب ماتریس 2×2 : ماتریس های M و N را ایجاد کردیم و با دستور زیر در Hadoop قرار دادیم:

Hadoop fs -copyFromLocal /home/ali/M.txt /Matrix

Hadoop fs -copyFromLocal /home/ali/N.txt /Matrix

```
h-user@master:~$ hadoop fs -cat /Matrix/M.txt
2020-12-30 14:47:53,914 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
M,0,0,1
M,0,1,2
M,1,0,3
M,1,1,4
h-user@master:~$ hadoop fs -cat /Matrix/N.txt
2020-12-30 14:48:03,515 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHostTrusted = false, remoteHostTrusted = false
N,0,0,5
N,0,1,6
N,1,0,7
N,1,1,8
```

برنامه ضرب ماتریس ما شامل سه فایل Map.java، Reduce.java و MatrixMultiply.java بود که مطابق روش گفته فایل jar را ساختیم و دو ماتریس را در یکدیگر ضرب کردیم:

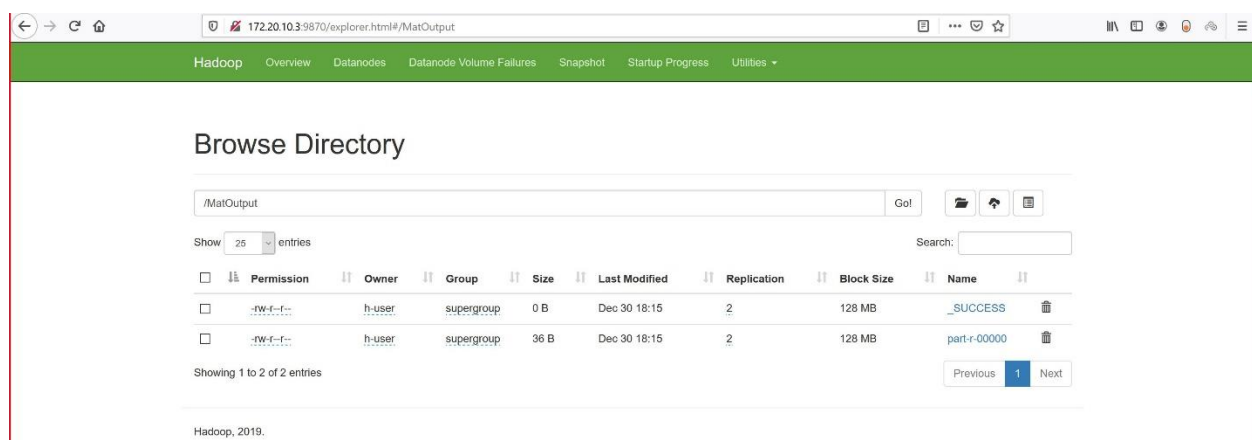
```
$ hadoop com.sun.tools.javac.Main Map.java Reduce.java MatrixMultiply.java
$ jar cf myp.jar Map.class Reduce.class MatrixMultiply.class
```

با دستور `hadoop jar myp.jar MatrixMultiply /Matrix /Matoutput` برنامه را اجرا کردیم.

بررسی صحت اجرا:

```
h-user@master:~$ hadoop fs -ls -R /
drwxr-xr-x - h-user supergroup          0 2020-12-30 14:45 /MatOutput
-rw-r--r-- 2 h-user supergroup          0 2020-12-30 14:45 /MatOutput/_SUCCESS
-rw-r--r-- 2 h-user supergroup        36 2020-12-30 14:45 /MatOutput/part-r-00000
drwxr-xr-x - h-user supergroup          0 2020-12-30 13:37 /Matrix
-rw-r--r-- 2 h-user supergroup        32 2020-12-30 13:36 /Matrix/M.txt
-rw-r--r-- 2 h-user supergroup        32 2020-12-30 13:37 /Matrix/N.txt
drwxr-xr-x - h-user supergroup          0 2020-12-30 14:01 /operation
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:07 /user
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:31 /user/hadoop
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:27 /user/hadoop/input
-rw-r--r-- 2 h-user supergroup       82 2020-12-26 15:27 /user/hadoop/input/test.txt
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:31 /user/hadoop/output
-rw-r--r-- 2 h-user supergroup          0 2020-12-26 15:31 /user/hadoop/output/_SUCCESS
-rw-r--r-- 2 h-user supergroup       58 2020-12-26 15:31 /user/hadoop/output/part-r-00000
h-user@master:~$ hadoop fs -cat /MatOutput/part-r-00000
2020-12-30 14:46:19,563 INFO sas1.SaslDataTransferClient: SASL encryption trust check: localHostTru
ted = false, remoteHostTrusted = false
0,0,19.0
0,1,22.0
1,0,43.0
1,1,50.0
```

مشاهده در webgui:



در ادامه ماتریس 400×400 (ماتریس 500×500 تا 84 درصد عملیات map را انجام می‌داد) را امتحان کردیم (فایل ماتریس ها ضمیمه شده اند). دستورات مانند ماتریس 2×2 بود فقط نحوه ورودی دادن آن به شکل زیر بود:

```
hadoop jar myp.jar MatrixMultiply /mat400 /output400
```

بررسی صحت اجرا:

```

h-user@master:~$ hadoop fs -ls -R /
drwxr-xr-x - h-user supergroup          0 2020-12-30 14:45 /MatOutput
-rw-r--r-- 2 h-user supergroup          0 2020-12-30 14:45 /MatOutput/_SUCCESS
-rw-r--r-- 2 h-user supergroup        36 2020-12-30 14:45 /MatOutput/part-r-00000
drwxr-xr-x - h-user supergroup          0 2020-12-30 22:03 /mat300
-rw-r--r-- 2 h-user supergroup    1104000 2020-12-30 22:03 /mat300/M300.txt
-rw-r--r-- 2 h-user supergroup    1104000 2020-12-30 22:03 /mat300/N300.txt
drwxr-xr-x - h-user supergroup          0 2021-01-04 19:04 /mat400
-rw-r--r-- 2 h-user supergroup    1992000 2021-01-04 19:03 /mat400/M400.txt
-rw-r--r-- 2 h-user supergroup    1992000 2021-01-04 19:04 /mat400/N400.txt
drwxr-xr-x - h-user supergroup          0 2020-12-30 14:01 /operation
drwxr-xr-x - h-user supergroup          0 2021-01-04 19:30 /output400
-rw-r--r-- 2 h-user supergroup          0 2021-01-04 19:30 /output400/_SUCCESS
-rw-r--r-- 2 h-user supergroup    2312002 2021-01-04 19:30 /output400/part-r-00000
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:07 /user
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:31 /user/hadoop
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:27 /user/hadoop/input
-rw-r--r-- 2 h-user supergroup         82 2020-12-26 15:27 /user/hadoop/input/test.txt
drwxr-xr-x - h-user supergroup          0 2020-12-26 15:31 /user/hadoop/output
-rw-r--r-- 2 h-user supergroup          0 2020-12-26 15:31 /user/hadoop/output/_SUCCESS
-rw-r--r-- 2 h-user supergroup         58 2020-12-26 15:31 /user/hadoop/output/part-r-00000
h-user@master:~$

```

Hadoop fs -cat /output400/part-r-00000:

```

349,258,8119.0
349,259,7560.0
349,26,8220.0
349,260,8116.0
349,261,7617.0
349,262,7375.0
349,263,8111.0
349,264,8775.0
349,265,8088.0
349,266,7641.0
349,267,7679.0
349,268,8724.0
349,269,7587.0
349,27,7948.0
349,270,7795.0
349,271,7909.0
349,272,8385.0
349,273,7813.0
349,274,7872.0
349,275,7830.0
349,276,8325.0
349,277,8217.0
349,278,7789.0
349,279,8601.0
349,28,7511.0
349,280,8237.0
349,280,8237.0
349,281,7977.0
349,282,7847.0
349,283,8126.0
349,284,8062.0
349,285,7277.0
349,286,7631.0
349,287,8457.0
349,288,7801.0
349,289,7805.0
349,29,8151.0

```

مشاهده در webgui:

→ ↻ 🏠 172.20.10.3:9870/explorer.html#/

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities ▾

Browse Directory

/

Go! 📁 ⬆️ 📄

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	drwxr-xr-x	h-user	supergroup	0 B	Dec 30 18:15	0	0 B	MatOutput	🗑️
<input type="checkbox"/>	drwxr-xr-x	h-user	supergroup	0 B	Dec 31 01:33	0	0 B	mat300	🗑️
<input type="checkbox"/>	drwxr-xr-x	h-user	supergroup	0 B	Jan 04 22:34	0	0 B	mat400	🗑️
<input type="checkbox"/>	drwxr-xr-x	h-user	supergroup	0 B	Dec 30 17:31	0	0 B	operation	🗑️
<input type="checkbox"/>	drwxr-xr-x	h-user	supergroup	0 B	Jan 04 23:00	0	0 B	output400	🗑️
<input type="checkbox"/>	drwxr-xr-x	h-user	supergroup	0 B	Dec 26 18:37	0	0 B	user	🗑️

Showing 1 to 6 of 6 entries

Previous 1 Next

Hadoop, 2019.

← → ↻ 🏠 172.20.10.3:9870/explorer.html#/output400

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities ▾

Browse Directory

/output400

Go! 📁 ⬆️ 📄

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	
<input type="checkbox"/>	-rw-r--r--	h-user	supergroup	0 B	Jan 04 23:00	2	128 MB	_SUCCESS	🗑️
<input type="checkbox"/>	-rw-r--r--	h-user	supergroup	2.2 MB	Jan 04 23:00	2	128 MB	part-r-00000	🗑️

Showing 1 to 2 of 2 entries

Previous 1 Next

Hadoop, 2019.

زمان اجرای برنامه برای ما با سیستم با مشخصات زیر در حدود ۲۵ دقیقه بود و مصرف پردازنده و حافظه ما طبق عکس زیر بود:

