

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

**Навчально-науковий інститут атомної та теплової енергетики
Кафедра цифрових технологій в енергетиці**

ЗВІТ

з лабораторної роботи №8

**з дисципліни «Розробка застосунків інтернету речей та
сенсорних мереж»**

**Тема: «Програмна платформа Apache Hadoop.
Використання шаблонів проєктування Mapreduce.
Алгоритм»**

Варіант №17

Виконав:

Студент групи ТР-12

Ковальов Олександр Олексійович

Дата здачі: 11.03.2025

Мета роботи. Розробка моделі кластеризації даних з використанням технології MapReduce на базі програмної платформи Apache Hadoop.

Індивідуальне завдання:

1. Постановка задачі кластеризації для всіх алгоритмів полягає в такому:
Дано: X – простір об'єктів; $XI = \{x_i\} \mid i=1$ — вибірка елементів;
 $d: X \times X \rightarrow [0; \infty)$ — функція відстані між об'єктами.
Знайти: Y – множину кластерів і відображення $a: X \rightarrow Y$ – алгоритм кластеризації такий, що кожен кластер складається з близьких між собою об'єктів, а об'єкти різних кластерів суттєво відрізняються.
2. Створити модель кластеризації даних за допомогою технології MapReduce.

Хід роботи.

Був встановлений Maven:

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.5487]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Alex>mvn -version
Apache Maven 3.9.9 (8e8579a9e76f7d015ee5ec7bfc9d260186937)
Maven home: C:\Program Files\Apache\Maven-3.9.9
Java version: 23.0.1, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk-23
Default locale: en_US, platform encoding: UTF-8
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\Users\Alex>
```

Також, був встановлений JDK 23.0.1:

```
C:\Users\Alex>java --version
java 23.0.1 2024-10-15
Java(TM) SE Runtime Environment (build 23.0.1+11-39)
Java HotSpot(TM) 64-Bit Server VM (build 23.0.1+11-39, mixed mode, sharing)

C:\Users\Alex>javac --version
javac 23.0.1

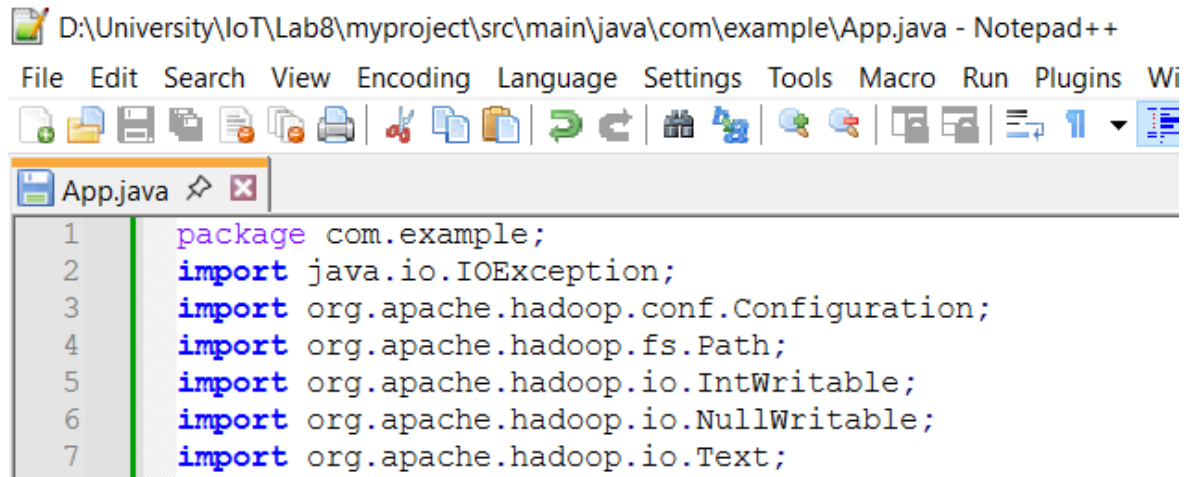
C:\Users\Alex>
```

Згенерований шаблон проекту:

```
[INFO] -----
[INFO] Using following parameters for creating project from Archetype: maven-archetype-quickstart:1.4
[INFO] -----
[INFO] Parameter: groupId, Value: com.example
[INFO] Parameter: artifactId, Value: myproject
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Parameter: package, Value: com.example
[INFO] Parameter: packageInPathFormat, Value: com/example
[INFO] Parameter: package, Value: com.example
[INFO] Parameter: groupId, Value: com.example
[INFO] Parameter: artifactId, Value: myproject
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Project created from Archetype in dir: D:\University\???\Lab8\myproject
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 18.419 s
[INFO] Finished at: 2025-03-10T22:29:11+02:00
[INFO] -----

D:\University\IoT\Lab8>
```

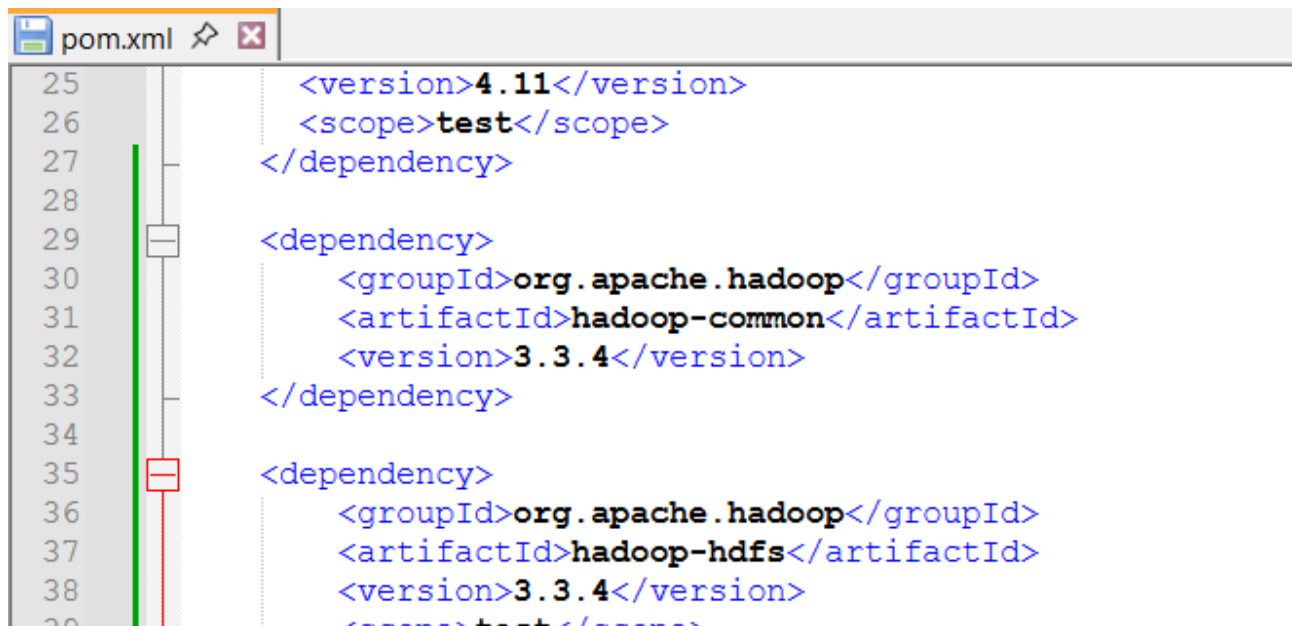
Заповнений файл з кодом:



The screenshot shows the Notepad++ application window titled "D:\University\IoT\Lab8\myproject\src\main\java\com\example\App.java - Notepad++". The menu bar includes File, Edit, Search, View, Encoding, Language, Settings, Tools, Macro, Run, Plugins, and Windows. The toolbar contains various icons for file operations and editing. The editor window shows the following Java code:

```
1 package com.example;  
2 import java.io.IOException;  
3 import org.apache.hadoop.conf.Configuration;  
4 import org.apache.hadoop.fs.Path;  
5 import org.apache.hadoop.io.IntWritable;  
6 import org.apache.hadoop.io.NullWritable;  
7 import org.apache.hadoop.io.Text;
```

Додані залежності:



The screenshot shows the pom.xml file in an IDE. The editor window displays the following XML code:

```
25 <version>4.11</version>  
26 <scope>test</scope>  
27 </dependency>  
28  
29 <dependency>  
30 <groupId>org.apache.hadoop</groupId>  
31 <artifactId>hadoop-common</artifactId>  
32 <version>3.3.4</version>  
33 </dependency>  
34  
35 <dependency>  
36 <groupId>org.apache.hadoop</groupId>  
37 <artifactId>hadoop-hdfs</artifactId>  
38 <version>3.3.4</version>  
39 <scope>test</scope>
```

Додані плагіни:



The screenshot shows the pom.xml file in an IDE, focusing on the build section. The editor window displays the following XML code:

```
49 <build>  
50 <pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->  
51 <plugins>  
52 <plugin>  
53 <artifactId>maven-assembly-plugin</artifactId>  
54 <executions>  
55 <execution>  
56 <phase>package</phase>  
57 <goals>  
58 <goal>single</goal>  
59 </goals>  
60 </execution>  
61 </executions>  
62 <configuration>  
63 <archive>  
64 <manifest>  
65 <addClasspath>true</addClasspath>  
66 <mainClass>com.example.App</mainClass>  
67 </manifest>  
68 </archive>  
69 </configuration>  
</plugin>  
</plugins>  
</pluginManagement>  
</build>
```

Маємо згенерований .jar файл:

```
.jar (318 KB at 1.2 MB/s)
[INFO] Building jar: D:\University\???\Lab8\myproject\target\myproject-1.0-SNAPSHOT-jar-with-dependencies.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 16.392 s
[INFO] Finished at: 2025-03-10T22:47:00+02:00
[INFO] -----
D:\University\IoT\Lab8\myproject>
```

Був встановлений дистрибув Ubuntu, WSL2:

```
xairaven@DESKTOP-6U4KUBG: ~
xairaven@DESKTOP-6U4KUBG:~$ echo "Lab 8, IoT. Alex Kovalov, 17"
Lab 8, IoT. Alex Kovalov, 17
xairaven@DESKTOP-6U4KUBG:~$ uname -ar
Linux DESKTOP-6U4KUBG 5.15.167.4-microsoft-standard-WSL2 #1 SMP Tue Nov 5 00:21:55 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
xairaven@DESKTOP-6U4KUBG:~$
```

I Docker Desktop:



Та git:

```
MINGW64:/c/Users/Alex
Alex@DESKTOP-6U4KUBG MINGW64 ~
$ git --version
git version 2.47.1.windows.1
Alex@DESKTOP-6U4KUBG MINGW64 ~
$ |
```

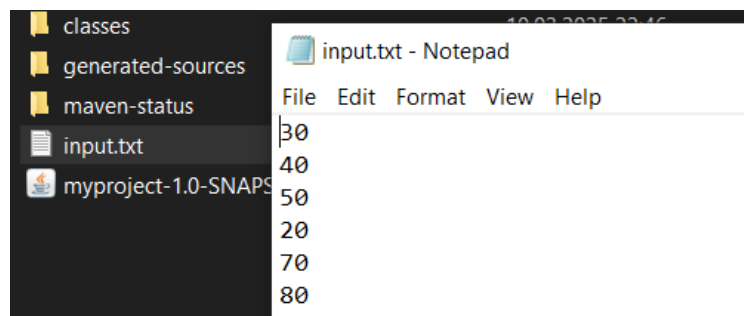
Був сконований проект:

```
D:\University\IoT\Lab8>git clone https://github.com/big-data-europe/docker-hadoop
Cloning into 'docker-hadoop'...
remote: Enumerating objects: 539, done.
remote: Counting objects: 100% (189/189), done.
remote: Compressing objects: 100% (23/23), done.
remote: Total 539 (delta 169), reused 166 (delta 166), pack-reused 350 (from 1)
Receiving objects: 100% (539/539), 108.00 KiB | 1.30 MiB/s, done.
Resolving deltas: 100% (251/251), done.
D:\University\IoT\Lab8>
```

Запущений контейнер:

<input type="checkbox"/>	Name	Container ID	Image	Port(s)
<input type="checkbox"/>	> docker-hadoop	-	-	-

Був створений файл «input.txt»:



Файли були скопійовані в контейнер:

```
D:\University\IoT\Lab8\myproject\target>docker cp myproject-1.0-SNAPSHOT-jar-with-de
Successfully copied 53.6MB to namenode:/tmp/

D:\University\IoT\Lab8\myproject\target>docker cp input.txt namenode:/tmp/
Successfully copied 2.05kB to namenode:/tmp/

D:\University\IoT\Lab8\myproject\target>_
```

Після входу в контейнер, в ньому були створені відповідні каталоги та скопійований вхідний файл:

```
D:\University\IoT\Lab8\myproject\target>docker exec -it namenode /bin/bash
root@aa2cfc7f008f:/# hdfs dfs -mkdir -p /user/root
root@aa2cfc7f008f:/# hdfs dfs -mkdir /user/root/input
root@aa2cfc7f008f:/# cd tmp/
root@aa2cfc7f008f:/tmp# hdfs dfs -put input.txt /user/root/input
2025-03-10 21:05:13,973 INFO sasl.SaslDataTransferClient: SASL encryption trust check:
ostTrusted = false
root@aa2cfc7f008f:/tmp#
```

Був запущений Hadoop:

```
root@aa2cfc7f008f:/tmp# hadoop jar myproject-1.0-SNAPSHOT-jar-with-dependencies.jar input output
2025-03-10 21:07:28,280 INFO client.RMPProxy: Connecting to ResourceManager at resourcemanager/172.18.0.4:
2025-03-10 21:07:28,389 INFO client.AHSPProxy: Connecting to Application History server at historyserver/1
2025-03-10 21:07:28,514 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not perfor
the Tool interface and execute your application with ToolRunner to remedy this.
2025-03-10 21:07:28,529 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoo
root/.staging/job_1741640351494_0001
```

Насамкінець, були виведені результати:

```
root@aa2cfc7f008f:/tmp# hdfs dfs -cat output/part-r-00000
2025-03-10 21:08:45,689 INFO sasl.SaslDataTransferClient: SASL encryption trust
ostTrusted = false
greater than 60 2
less than or equal to 60 4
root@aa2cfc7f008f:/tmp# echo "Alex Kovalov"
Alex Kovalov
root@aa2cfc7f008f:/tmp# echo "Variant 17"
Variant 17
root@aa2cfc7f008f:/tmp#
```

Висновок: У ході виконання лабораторної роботи була розроблена модель кластеризації даних із використанням технології MapReduce на базі Apache Hadoop. Було налаштовано середовище розробки, включаючи встановлення необхідних компонентів, таких як JDK, Maven, Docker та Hadoop. Було реалізовано алгоритм кластеризації, який дозволив розподілити вхідні дані на окремі групи відповідно до визначених критеріїв. Отримані результати підтвердили коректність виконання алгоритму та правильність роботи системи MapReduce. Таким чином, виконана робота продемонструвала можливості Apache Hadoop у розподіленій обробці великих обсягів даних і підтвердила ефективність використання даної технології для задач кластеризації.

Контрольні питання:

1. *Які ключові завдання стоять перед задачею кластеризації даних за допомогою технології MapReduce на базі Apache Hadoop?*

Ключовими завданнями кластеризації даних за допомогою технології MapReduce на базі Apache Hadoop є розподіл великих обсягів даних на однорідні групи, ефективна обробка цих даних у розподіленому середовищі, а також забезпечення масштабованості та швидкодії процесу кластеризації.

2. *Яким чином функціонує технологія MapReduce у розв'язанні завдань аналізу великих обсягів даних, і чи є вона оптимальною для пошукових завдань?*

Технологія MapReduce розподіляє обробку великих обсягів даних між кількома вузлами кластера, розбиваючи задачу на дві основні стадії: Map (перетворення вхідних даних у проміжні ключ-значення) і Reduce (агрегування та обробка проміжних результатів). Для пошукових завдань MapReduce може бути ефективною, якщо проблема добре розподіляється на

незалежні підзадачі, але для складних пошукових операцій інші технології, такі як Spark, можуть бути більш оптимальними.

3. *Які переваги та особливості використання Apache Hadoop у контексті обробки і зберігання великих обсягів даних? Які основні компоненти Apache Hadoop?*

Apache Hadoop має кілька ключових переваг: можливість обробки великих обсягів даних, висока відмовостійкість, горизонтальна масштабованість і ефективність роботи з неструктурованими даними. Основні компоненти Hadoop – це HDFS (розподілена файлова система для зберігання даних), YARN (менеджер ресурсів), MapReduce (модель обробки даних) і Hadoop Common (базові бібліотеки та утиліти).

4. *Яким чином розподілена файлова система HDFS забезпечує надійність і відмовостійкість у системі Hadoop?*

HDFS забезпечує надійність і відмовостійкість завдяки розподіленому зберігання даних, реплікації блоків файлів на кількох вузлах і автоматичному відновленню при збоях. Якщо один вузол виходить з ладу, дані залишаються доступними на інших вузлах.

5. *Як працює функціонал MapReduce на етапі розділення та складання, які взаємодії відбуваються між вузлами кластера?*

На етапі розділення (Map) дані розбиваються на фрагменти та обробляються незалежними вузлами, які створюють пари ключ-значення. Далі ці проміжні результати передаються на вузли, які виконують фазу складання (Reduce), де дані агрегуються та обробляються відповідно до алгоритму. Взаємодія між вузлами здійснюється через HDFS, що забезпечує збереження даних і розподіл навантаження.

6. *Які основні кроки включає алгоритм кластеризації за схемою K-середніх, як відзначено в матеріалі?*

Алгоритм кластеризації за схемою K-середніх включає кілька основних кроків: вибір початкових центрів кластерів, обчислення відстані між кожним об'єктом і центрами кластерів, призначення об'єктів до найближчого кластера, обчислення нових центрів кластерів та повторення процесу до збіжності.

7. *Яким чином виправдано використання Apache Hadoop для обробки великих обсягів даних у завданнях кластеризації, зокрема, які переваги це надає в контексті масштабованості та розподіленої обробки даних?*

Використання Apache Hadoop для обробки великих обсягів даних у задачах кластеризації виправдане завдяки можливості паралельної та розподіленої обробки. Hadoop дозволяє обробляти великі дані навіть на недорогому апаратному забезпеченні, забезпечує надійність, високу масштабованість та ефективність роботи з різними типами даних.