Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Ордена Трудового Красного знамени

Федеральное государственное бюджетное учреждение

высшего образования

«Московский технический университет связи и информатики»

Кафедра МКиТ

Лабораторная работа № 8

«Веб-сканнер»

Выполнил студент

Группы БСТ1903

Самоваров А.П.

Вариант № 12

Москва 2021

**Оглавление**

[1. Цель лабораторной работы 3](#_Toc72366484)

[2. Задание на лабораторную работу 3](#_Toc72366485)

[3. Код программы 3](#_Toc72366486)

[4. Результат работы программы 7](#_Toc72366487)

[Вывод 8](#_Toc72366488)

1. Цель лабораторной работы

Реализовать простой веб-сканнер, который будет автоматически загружать веб-страницы из Интернета, искать на этих страницах новые ссылки и повторять их. Реализовать многопоточность.

2. Задание на лабораторную работу

Написать программу, которая выполняет функции веб-сканнера.

3. Код программы

Код для класса Crawler (в данной работе меняется только данный класс):

import java.io.\*;

import java.net.\*;

import java.util.\*;

import java.util.regex.\*;

class Crawler {

private HashMap<String, URLDepthPair> links = new HashMap<>();

private LinkedList<URLDepthPair> pool = new LinkedList<>();

private int depth = 0;

public Crawler(String url, int depth\_) {

depth = depth\_;

pool.add(new URLDepthPair(url, 0));

}

public void run() {

while (pool.size() > 0 || Thread.activeCount() > 1) {

if (pool.size() > 0) {

URLDepthPair link = pool.pop();

CrawlerThread task = new CrawlerThread(link);

task.start();

}

}

for (URLDepthPair link : links.values())

System.out.println(link);

System.out.println();

System.out.printf("Found %d URLS\n", links.size());

}

public static Pattern LINK\_REGEX = Pattern.compile(

"<a\\s+(?:[^>]\*?\\s+)?href=([\"'])(.\*?)\\1"

);

private class CrawlerThread extends Thread {

private URLDepthPair link;

public CrawlerThread(URLDepthPair link\_) {

link = link\_;

}

@Override

public void run() {

if (links.containsKey(link.getURL())) {

URLDepthPair knownLink = links.get(link.getURL());

knownLink.incrementVisited();

return;

}

links.put(link.getURL(), link);

if (link.getDepth() >= depth)

return;

try {

URL url = new URL(link.getURL());

HttpURLConnection con = (HttpURLConnection) url.openConnection();

con.setRequestMethod("GET");

Scanner s = new Scanner(con.getInputStream());

while (s.findWithinHorizon(LINK\_REGEX, 0) != null) {

String newURL = s.match().group(2);

if (newURL.startsWith("/"))

newURL = link.getURL() + newURL;

else if (!newURL.startsWith("http"))

continue;

URLDepthPair newLink = new URLDepthPair(newURL, link.getDepth() + 1);

pool.add(newLink);

}

} catch (Exception e) {}

}

}

public static void showHelp() {

System.out.println("usage: java Crawler <URL> <depth>");

System.exit(1);

}

public static void main(String[] args) {

if (args.length != 2) showHelp();

String url = args[0];

int depth = 0;

try {

depth = Integer.parseInt(args[1]);

} catch (Exception e) {

showHelp();

}

Crawler crawler = new Crawler(url, depth);

crawler.run();

}

} Код для класса URLDepthPair:

class URLDepthPair {

private String url;

private int depth;

private int visited;

public URLDepthPair(String url\_, int depth\_) {

url = url\_;

depth = depth\_;

visited = 1;

}

public String getURL() {

return url;

}

public int getDepth() {

return depth;

}

public void incrementVisited() {

visited++;

}

public String toString() {

return "<URL href=\"" + url + "\" visited=\"" + visited + "\" depth=\"" + depth + "\" \\>";

}

}

4. Результат работы программы

На рисунке 1 представлен запуск программы через консоль.

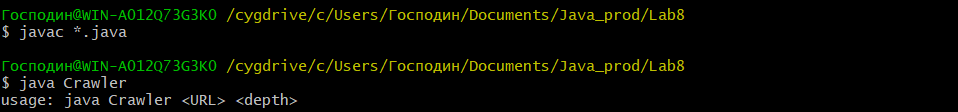


Рисунок 1 – Запуск программы через консоль

На рисунке 2 представлена первая проверка работы программы.

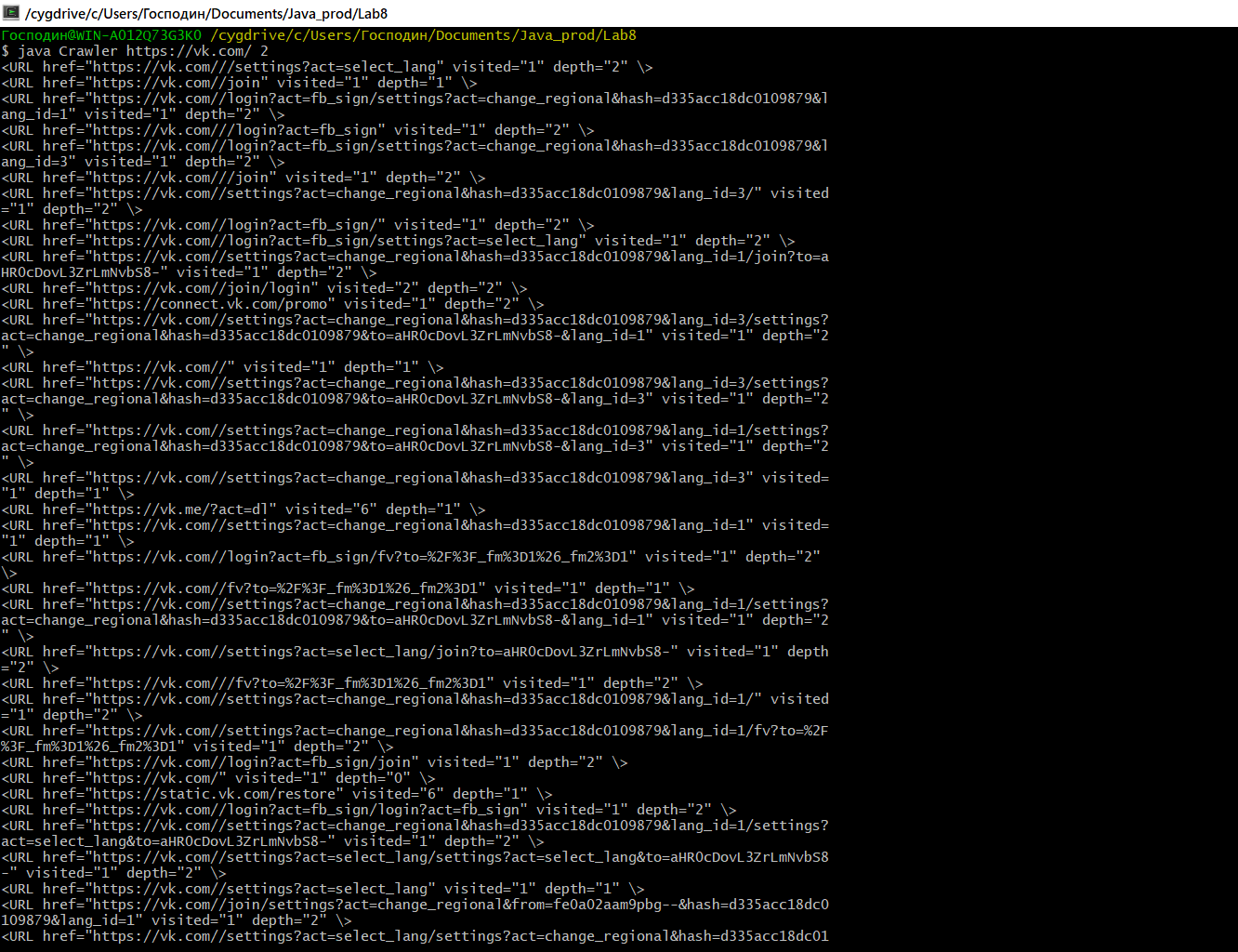


Рисунок 2 – Первая часть тестирования программы

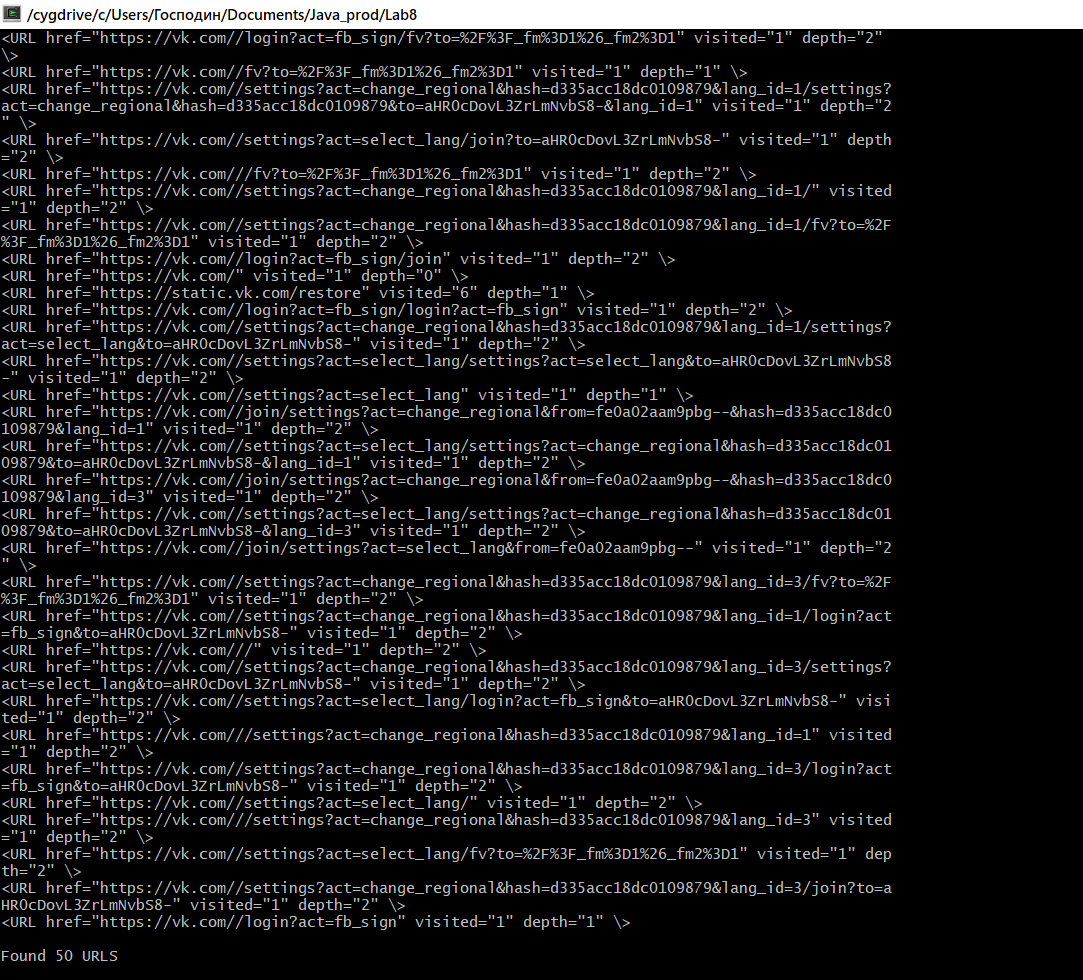


Рисунок 2 – Вторая часть тестирования программы

Вывод

В ходе выполнения данной лабораторной работы были освоены навыки работы с некоторыми пакетами в Java (в частности с java.io.\* и java.net.\*). А также написана программа, которая выполняет функции простого веб-сканнера, также реализована многопоточность.