Министерство цифрового развития, связи и

Массовых коммуникаций Российской Федерации

Ордена Трудового Красного Знамени

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра «Математической кибернетики и информационных технологий»

Отчет по лабораторной работе №7

по дисциплине «Кроссплатформенные технологии программирования»

Выполнил: студент группы БСТ2001

Багдасарян Артём

Москва 2022

Оглавление

[**1.** **Цель и задание** 3](#_Toc100097405)

[**2.** **Ход выполнения лабораторной работы** 3](#_Toc100097406)

[**2.1** **Код программы** 3](#_Toc100097407)

[**2.2** **Результат работы программы** 5](#_Toc100097408)

[**3.** **Вывод** 5](#_Toc100097409)

[**Список использованных источников** 6](#_Toc100097410)

1. **Цель и задание**

В данной лабораторной работе мы создадим элементарный веб-сканер.

1. **Ход выполнения лабораторной работы** 
   1. **Код программы**

Ниже представлен код Crawler.java

package com.company;  
  
import java.net.HttpURLConnection;  
import java.net.URL;  
import java.util.HashMap;  
import java.util.LinkedList;  
import java.util.Scanner;  
import java.util.regex.Pattern;  
  
class Crawler<LINK\_REGEX> {  
 private HashMap<String, URLdepthPair> links = new HashMap<>();  
 private LinkedList<URLdepthPair> pool = new LinkedList<>();  
  
 private int depth = 0;  
  
 public Crawler(String url, int dep) {  
 depth = dep;  
 pool.add(new URLdepthPair(url, 0));  
 }  
  
 public void run() {  
 while (pool.size() > 0)  
 parseLink(pool.pop());  
  
 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 void parseLink(URLdepthPair link) {  
 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){  
 Scanner scan=new Scanner(System.*in*);  
 String[] argg = new String[2];  
 System.*out*.println("Enter URL: ");  
 argg[0]=scan.nextLine();  
 System.*out*.println("Enter depth: ");  
 argg[1]=scan.nextLine();  
 if (argg.length !=2) *showHelp*();  
 int depth = 0;  
 String url=argg[0];  
 try {  
 depth = Integer.*parseInt*(argg[1]);  
 } catch (Exception e) {  
 *showHelp*();  
 }  
 Crawler crawler = new Crawler(url, depth);  
 crawler.run();  
 }

}

Ниже представлен код FractalExplorer.java

package com.company;  
  
public class URLdepthPair {  
 private String url;  
 private int depth;  
 private int visited;  
 public URLdepthPair(String URL, int dep) {  
 url = URL;  
 depth=dep;  
 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 + "\" \\>";  
 }  
}

* 1. **Результат работы программы**

Результат работы программы показан на рисунке 1

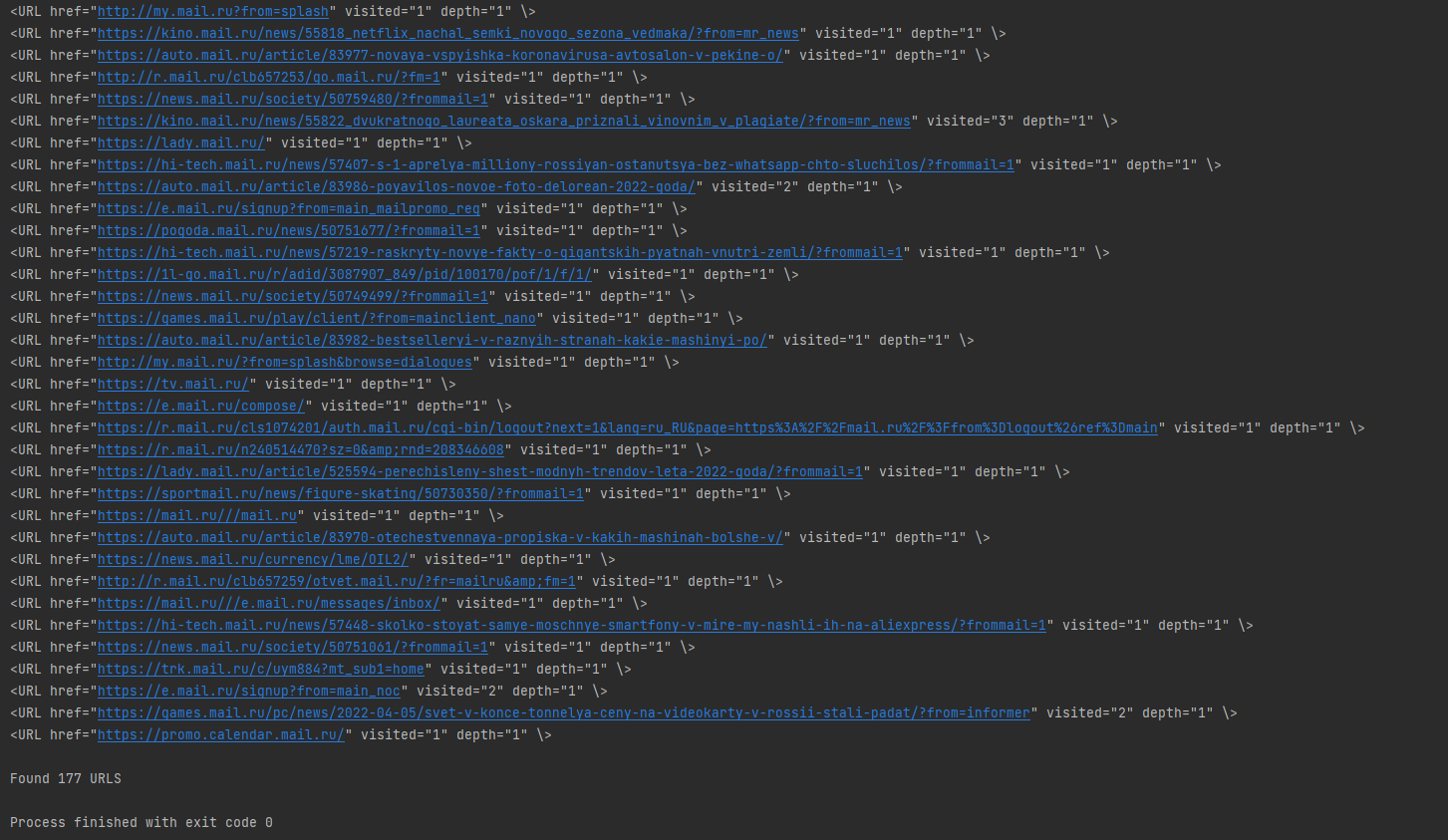


Рисунок 1 – Результат работы программы

* 1. **Программа на Git hub**

Ссылка на репозиторий на Git hub: https://github.com/Artem2406/KTP

1. **Вывод**

В данной лабораторной работе мы создали элементарный веб-сканер.

# **Список использованных источников**

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа, 2006.

2. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004.