University of Leeds **School of Computing**
Web Services and Web Data COMP3011
Semester 2, 2021-2022

Coursework 2 - Building a Search Tool
30 marks = 30% of total module marks

Important Note: This is an individual project, NOT a team project. Each student must implement their own search tool.

Submission Deadline: **6/5/2021 at 11:00 pm (UK time)**

In this project, you will develop a search tool that can:

1) Crawl the pages of a website.

2) Create an inverted index of all word occurrences in the pages of a website.

3) Allow the user to find pages containing certain search terms.

The website you will use for this project is http://example.python-scraping.com/ . This website contains brief information about each country in the world, such as its capital, area, population, currency, … etc. The website was purpose-built to allow students to learn web scraping. I have obtained permission from the website's owner for us to crawl and download the pages of the website. However, you must observe a politeness window of at least 5 seconds between successive requests to the website. An inverted index that stores the frequency of occurrence of each word in each page must be created by the tool as it crawls the pages of the website.

Using the search tool, the user should be able to find pages containing individual words such as 'Mariehamn', or a combination of two or more words such as 'Capital Aland Islands'.

The search tool is to be command-driven and must provide the following commands:

build

This command instructs the search tool to crawl the website, build the index, and save the resulting index into the file system. For simplicity you can save the entire index in one file.

load

This command loads the index from the file system. Obviously, this command will only work if the index has previously been created using the 'build' command.

print

This command prints the inverted index for a particular word, for example:

print Peso

will print the inverted index for the word 'Peso'

find

This command is used to find a certain query phrase in the inverted index and returns a list of all pages containing this phrase, for example:

find Dinar

will return a list of all pages containing the word 'Dinar', while

find Area Afghanistan

will return all pages containing the words 'Area' and 'Afghanistan'.

For simplicity assume that the search is case sensitive, so 'Euro' is not the same word as 'euro'.

You should use Python 3 to implement the search tool. It is also strongly recommended to you use the 'Requests' library (http://docs.python-requests.org/en/master/) for composing requests, and the 'Beautiful Soup' library (https://www.crummy.com/software/BeautifulSoup/bs4/doc/) to parse HTML pages.

To submit the source code of your search tool to Minerva, put your Python source file(s) and the inverted index file that was created by your tool in one directory, compress the directory with ZIP, and upload it to Minerva. As part of your submission, you should also submit a brief report (2-3 pages excluding the title page) that clearly, yet briefly, describes how you implemented each aspect of the tool. For example, the data structures, methods and algorithms you have used in 1) crawling the website, 2) creating the inverted index, and 3) computing the scores of pages when processing a search query. The report should also include brief instructions on how to invoke and use the tool. Please do NOT fill your report by copying text from online resources, such as tutorials or lecture slides, as I am only interested to understand what you have done yourself in this coursework.

Marking Scheme

The tool successfully crawls all the pages of the website	(6 marks)
The tool successfully creates the inverted index for the whole website	(6 marks)
The tool can store then load the inverted index to/from the file system	(6 marks)
The tool prints the inverted list for a certain word	(4 marks)
The tool can correctly find pages containing search terms	(8 marks)

The clarity of your report will affect the marks you are awarded for the relevant aspects of the mark scheme.