

**NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY**

School of Electrical Engineering and Computer Sciences

**WEB ENGINEERING**

**ASSIGNMENT# 2**

**SUBMITTED BY: Umaamah Leghari**

**CMS ID: 382799**

**CLASS + SECTION: BESE 12 A**

**DOCUMENTATION:**

Git repository link: <https://github.com/ULeghari/web_crawler>

**HOW TO USE:**

1. Go to xampp control panel and Start Apache.
2. Then open the crawler.php file in browser. This file will crawl, parse and save content in flat files as well as exract the URLs in the file. Wait for all URLs to be parsed till depth reached, or timeout occurs. This may take a minute.
3. Next open the searchResults.php file in the browser, and search for the content you want to find.

**DESCRIPTION:**

**crawler.php:**

It represents a web crawler that has a seed URL and recursively crawls other URLs in it to a specific depth limit.

**FUNCTIONALITIES:**

1. **URL QUEUE:**

The seed URL and maximum crawling depth are set.The URL queue is initialized as an array.The seed URL is then added to the queue with an initial depth of 0.

1. **Crawling Loop:**

The main crawling loop continues until the URL queue is empty.Each iteration dequeues a URL and retrieves its HTML content.If HTML content is successfully obtained and the current depth is within the specified limit, the title and meta description are extracted and displayed.The script then parses the HTML to extract URLs, and if the meta description is found, these URLs are added to the URL queue with an incremented depth.

1. **Error Handling:**

The script includes exception handling to catch any errors that might occur during crawling.Exceptions are echoed to the output for debugging purposes.

**functions.php:**

It contains the functions used by the web crawler to crawl, parse, and extract HTML information.

**FUNCTIONALITIES:**

1. **crawl Function:**

This function accepts a URL, a filename for storing HTML content, and the current depth. It thenfetches the HTML content of the URL using cURL, and then, the URL and HTML content are appended to the specified file. It returns the HTML content.

1. **parse Function:**

This function accepts HTML content, the base URL, and the current depth. It usesDOMDocument and XPath to parse the HTML and extract hyperlinks (**<a>** tags). It returns an array of URLs with associated depths.

1. **extractInformation Function:**

This function accepts HTML content, and usesDOMDocument to extract the title and meta description from the HTML.It returns an associative array containing the title and meta description.

1. **urljoin Function:**

This function accepts a base URL and a relative URL, and then combines the base URL and relative URL to create an absolute URL, and returns the absolute URL.

**searchResults.php:**

This script represents a basic search engine interface where users can input a search query. The script provides advanced search features like case sensitivity and regular expression search.

**FUNCTIONALITIES:**

1. **Search Form:**

A form is displayed with an input field for the search query (searchStr).Users can also choose to enable case sensitivity (caseSensitive) and regular expression search (useRegex) using checkboxes.

1. **Search Logic:**

Upon form submission, the script retrieves the search query and advanced search options (case sensitivity and regular expression).It iterates through text files (assumed to be crawled web pages) stored in the current directory.For each file, it reads the content and extracts the URL, title, and meta description.It checks if the search query is found in the content based on the specified search options.If a match is found, it displays the URL, title, and a truncated meta description in a formatted manner.

1. **Advanced Search Features:**
   * **Case Sensitivity:** If the case sensitivity option is selected, the script performs a case-sensitive search using strpos.
   * **Regular Expression Search:** If the regular expression option is selected, the script uses preg\_match to perform a regular expression search.
   * **Default Search:** If neither option is selected, a case-insensitive search is performed using stripos.
2. **Result Display:**

Matching results are displayed in a formatted manner, showing the title, a truncated meta description, and the URL.If no match is found, a message indicating the absence of a match is displayed.

1. **Integration with sample3.php:**

The script includes the necessary functions from **sample3.php** to extract information from HTML content.

1. **Styling:**

The script includes an external style sheet (**style.css**) for basic styling.

**REQUIREMENTS:**

**URL Queue:**

I have implemented an array, ‘$urlQueue’, that acts as a queue to store URLs. The seed URL is first pushed into the array. Once a URL is parsed, it is removed from the queue, and function is applied to next URL in the queue.

**Crawling:**

The ‘crawl’ function in the ‘functions.php’ file sends HTTP request to the URLs in the queue, retrieves the HTML content, and saves it in flat .txt files so that search functionality can be applied on them later.

**HTML Parsing:**

The ‘parse’ function in the ‘functions.php’ file parses the HTML content using DOMDocumentclasss of PHP and extracts the hyperlinks in the URL.

**URL Extraction:**

The ‘parse’ function in the ‘functions.php’ file extracts the hyperlinks and adds them to the URL queue.

**Depth Limit:**

Maximum depth of three has been set which is checked in the parse and crawl functions so that depth does not exceed 3.

**Output:**

The title and meta description information is extracted from each URL in the ‘extractInformation’ function in ‘functions.php’, and is echoed to print them on screen.

**Content Search Module:**

Content search module is implemented in the searchResult.php file, which searches for a string in the saved files.

**Robots.txt Compliance:**

Robots.txt file’s existence is checked in the crawl function.

**Error Handling:**

Error handling is done in the crawl function by checking for cURL errors.

**Persistent Storage:**

The crawled data is stored in flat files to make it persistent. Once the data is crawled, files are made, and the URLs do not need to be crawled again each time.

**Advanced Search Features:**

Advanced search features have been implemented in the searchResults.php file with checkboxes for case sensitivity and regular expression.