***OPERATING SYSTEM PROJECT***

*Project-1:*

***Multi-threaded Web Crawler:*** Implement a multi-threaded web crawler. The crawler should be able to remember the last URLs and able to resume. Your program should be able to create an appropriate number of threads.

*Program code in Python:*

import queue

import threadingimport requestsimport reimport timeimport csv# Global variablesmax\_threads = 10 # maximum number of threads to usevisited\_urls = set() # set to keep track of visited URLsurls\_to\_visit = queue.Queue() # queue to keep track of URLs to visitcsv\_file\_name = "crawled\_data.csv"# Function to retrieve the HTML content of a pagedef get\_html(url): try: response = requests.get(url, timeout=5) return response.text except: return ""# Function to extract all the links on a pagedef get\_links(html): links = re.findall(r'href=[\'"]?([^\'" >]+)', html) return links# Function to visit a URL and extract all the linksdef visit\_url(url): global visited\_urls, urls\_to\_visit html = get\_html(url) links = get\_links(html) visited\_urls.add(url) for link in links: if link not in visited\_urls: urls\_to\_visit.put(link) with open(csv\_file\_name, 'a', newline='') as f: writer = csv.writer(f) writer.writerow([url])# Function to create and start threads to visit URLsdef start\_threads(): global urls\_to\_visit threads = [] while not urls\_to\_visit.empty(): # Create a new thread for each URL num\_threads = min(max\_threads, urls\_to\_visit.qsize()) for i in range(num\_threads): url = urls\_to\_visit.get() thread = threading.Thread(target=visit\_url, args=(url,)) threads.append(thread) # Start the threads for thread in threads: thread.start() # Wait for the threads to finish for thread in threads: thread.join() # Clear the threads list for the next iteration threads.clear() # Save the list of visited URLs to a file with open("visited\_urls.txt", "w") as f: for url in visited\_urls: f.write(url + "\n")# Function to read the list of visited URLs from a filedef read\_visited\_urls(): global visited\_urls try: with open("visited\_urls.txt", "r") as f: visited\_urls = set(f.read().splitlines()) except FileNotFoundError: pass# Main functiondef main(): global urls\_to\_visit read\_visited\_urls() urls\_to\_visit.put("https://www.google.com") while not urls\_to\_visit.empty(): start\_time = time.time() start\_threads() print(f"Visited {len(visited\_urls)} URLs") print(f"Time elapsed: {time.time()-start\_time:.2f} seconds") time.sleep(5) # wait for 5 seconds before starting the next iterationif \_\_name\_\_== "\_\_main\_\_": main()

*\*\*THE END\*\**

By,

Sasank Tumma

CSE-D (AP21110010237)