De La Salle University
Computer Technology Department

STDISCM

Parallel Programming Project – Email Address Web Scraper

Description

An organization uses websites to disseminate information to potential customers or partners about the organization. An organization's website normally posts email addresses as contact information. Scraping email addresses each page manually in a website takes a long time. A web scraper is an automated tool that can scrape pages in the website. The web scraped can be programmed to automatically find email addresses using parallel programming techniques.

Project Requirement

The following are the requirements for the project:

- Create an email web scraper program that finds email addresses from a website in a specific amount of time
  - Input argument of the program are the following:
    - URL of the website to be scraped
    - Scraping time in minutes
    - Optional: Number of threads / process to use
  - Output of the program:
    - Text file that contains email and its associated name, office, department or unit in CSV format
    - Text file that contains statistics of the website: URL, number of pages scraped and number of email addresses found
- Website for scraping: https://www.dlsu.edu.ph
  - Do not use other websites
- Program implementation:
  - Program can be implemented using any programming language
  - Program implementation should use parallel programming techniques
  - Program implementation can use libraries or APIs that does web page retrieval
    - Libraries or APIs that does the scraping and finding automatically is not allowed

Project Rubrics

The project is to be graded using the following criteria / rubric:

CRITERIA	EXEMPLARY 4	SATISFACTORY 3	DEVELOPING 2	BEGINNING 1
Technical Documentation 10 %	Document has presented the architecture of the system, pointed out the concepts, has given an excellent analysis of the performance of the system and provided a conclusion.	Document has presented the architecture of the system, pointed out the concepts, has given simple analysis of the performance of the system.	Document has presented the architecture of the system and pointed out the concepts.	No documentation
Parallel Techniques 40 %	Multiple processes or threads are used to distribute workload by using synchronization or parallel techniques	Multiple processes / threads are used but workload was not distributed nor use parallel techniques	Program essentially uses serial programming techniques	No program submitted
Performance 50%	Project is able to achieve task and result of task is consistent		Project is able to achieve required task but not done in parallel manner or result is not consistent	Project is not working

Documentation

Documentation requirements for the project is as follows:

- Document should have the outline:
  1. Introduction
     - Give a brief discussion of the project and its requirement
  2. Program Implementation
     - Discussion on how the program was implemented
       - Use of lock or semaphore objects
       - Sharing of data between processes
       - Parallel programming and optimization techniques used
  3. Result
     - Discussion of the results
     - Explanation or analysis why such results was achieved
  4. Conclusion

     Discuss briefly how parallel programming techniques was used

     Discuss how parallel programming techniques improved (or not improved) performance
  5. References
     - References used for concepts, programming techniques or libraries used

- Document is to follow the IEEE manuscript template for conference proceeding
  - Format for the manuscript is found at: IEEE - Manuscript Templates for Conference Proceedings

Submission Requirements

For submission:

- Document report
- Program Source Code
- Program output file (Multiple samples to show performance)
- Screenshots (If needed)

Deadline: Week 9 – October 29, 2024 (Tentative)