**COMPLEX COMPUTING PROBLEM**

**(ASSIGNMENT#3)**

**OPERATING SYSTEMS**

|  |  |
| --- | --- |
| INSTRUCTOR | Ms. Laraib Razzaq |
| COURSE CODE | CSC342/CSE342/CSC222 |
| SEMESTER | 4th |
| CREDIT HOURS | 3 (Theory) + 1 (Lab) |
| KNOWLEDGE DOMAIN | COMPUTING CORE |
| Marks | 10 |

**Objective:**

The objective of this complex computing activity is to carry out research, analysis, design, investigation, and implementation of a real-world complex computing project that has the following attributes:

1. WP: Depth of Analysis Required: The activity requires abstract thinking and originality in analysis to formulate suitable computing models of the activity.
2. WP: Innovation: The activity involves creative use of computing principles and research-based knowledge in novel ways.
3. WP: Familiarity: The activity can extend beyond previous experiences by applying principle-based approaches.

**Complex Computing Activity Outcomes:**

The outcome of this activity is that the students may be able to partially attain some or all of the following graduate attributes:

1. GA1: Academic Education
2. GA2: Knowledge for Solving Computing Problems
3. GA3:Problem Analysis
4. GA4: Design/Development of Solutions

The activity is mapped on CLO2 and your performance in this project will play an important role in their attainment.

**Problem Statement:**

You are part of a team tasked with improving the performance of a virtual memory system for a modern operating system. The system currently uses **demand paging** and has the following configuration:

* **Page Size:** 4 KB
* **Total Physical Memory:** 16 MB
* **Virtual Memory:** 64 MB
* **Page Replacement Algorithm:** Least Recently Used (LRU)

Your task is to analyze the current system, simulate its performance under specific workloads, and propose enhancements to improve its efficiency.

#### **Tasks:**

* + Simulate a workload with 10 processes, each requesting memory between 5 MB and 10 MB.
  + Use a page reference string for each process to simulate memory access.
  + Implement and compare **LRU** with at least one other page replacement algorithm.
  + Evaluate the performance of algorithm by calculating:
    - Page fault rate for each algorithm.
    - Total memory effective access time, considering:
      * Disk access latency = 5 ms (for a page fault).
      * Memory access latency = 0.2 ms.

**Marking Rubrics:**

Projects were evaluated on the following criteria:

* Idea/Initial Study 20%
* Simulation 20%
* Report 60%