**NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES - FAST**



**Project Proposal CS-2006 Operating Systems**

**14/03/2023**

**Teacher: Dr Ghufran**

**Section: BS(CS)-4F**

**Semester: Spring 2023**

**Page replacement algorithm**

**Member 1: 21K-3267 Muhammad Saad**

**Member 2: 21K-3175 Syed Muhammad Taqi Baquer**

**Introduction**

Whenever a process is to be executed, it is brought into main memory. Only those pages of a process are brought into the main memory that is required. If the accessed page is not in the main memory, then it is known as page fault. Page replacement algorithms are used in computer operating systems to manage the allocation of memory to different processes. When a process needs to access a page of memory that is not currently in physical memory, a page fault occurs and the operating system must decide which page to evict from memory to make room for the requested page.

**Goal**

The goal of a page replacement algorithm is to minimize the number of page faults that occur, while also minimizing the overhead of managing the page table. There are many different page replacement algorithms, each with their own strengths and weaknesses. The page replacement algorithm includes FIFO, Optimal, LRU (Least Recently used),LFU(Least Frequently used), Clock and second chance. page replacement algorithms decide which memory pages to page out, sometimes called swap out, or write to disk, when a page of memory needs to be allocated. Page replacement happens when a requested page is not in memory (page fault) and a free page cannot be used to satisfy the allocation, either because there are none, or because the number of free pages is lower than some threshold.

**Methods**

There are many page replacement algorithm but we will be using these 3 page replacement algorithm which is as followed**:**

* First-In-First-Out (FIFO): This algorithm simply evicts the oldest page in memory when a page fault occurs.
* Least Recently Used (LRU): This algorithm evicts the least recently used page in memory when a page fault occurs.
* Optimal: This algorithm is a theoretical algorithm that always evicts the page that will not be used for the longest period of time in the future. This algorithm is not practical in real systems because it requires knowledge of future memory accesses.

**Tools**

1. C language
2. Ubuntu
3. System is 64 bit