作業系統-書面報告

資訊三甲 11127138 林雨臻

指導老師：鍾武君

1. **開發環境：**
2. **硬體設備**：：

* 處理器 (CPU)：11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHz
* 記憶體 (RAM)：16 GB
* 作業系統：Windows

1. **開發工具**：

* 編譯器：g++ (版本 12.2.0)
* 程式編輯器：vscode (Visual Studio Code)

1. **實驗方法和流程**
   1. FIFO\_Algorithm:

* 說明：當page frame已滿且需要替換時，移除佇列中的第一個進入的page。
* 資料結構：

    queue<int> pageQueue; // page frame

    unordered\_map<int, bool> Page\_Records; // page -> isExist

* 實作方法：
  1. LRU\_Algorithm:
* 說明：當page frame已滿且需要替換時，移除佇列中current\_time-最新使用點最大的page。
* 資料結構：

unordered\_map<int, int> pagePosition; // page -> timestamp

* 實作方法：find\_oldest()
  1. LFU\_and\_FIFO\_Algorithm:
* 說明：
* 資料結構：

unordered\_map<int, page\_record> pageRecords; // page -> (in\_page\_frame\_flag, count, timestamp)

set<pair<int, pair<int, int>>, Compare> pageFrames\_data; // <page, <count, timestamp>>

* 實作方法：find\_least\_frequent()
  1. MFU\_and\_FIFO\_Algorithm:
* 說明：
* 資料結構：

unordered\_map<int, page\_record> pageRecords; // page -> (in\_page\_frame\_flag, count, timestamp)

set<pair<int, pair<int, int>>, Compare> pageFrames\_data; // <page, <count, timestamp>>

* 實作方法：find\_most\_frequent()
  1. LFU\_and\_LRU\_Algorithm:
* 說明：
* 資料結構：

unordered\_map<int, page\_record> pageRecords; // page -> (in\_page\_frame\_flag, count, timestamp)

set<pair<int, pair<int, int>>, Compare> pageFrames\_data; // <page, <count, timestamp>>

* 實作方法：find\_least\_frequent\_and\_recently()

1. 探討結果和原因
   1. **方法一：**

**方法二：**

**方法三：**

**方法三：**