

作業二

- **問題描述：**於 **Linux 環境**，將作業一的程式碼變成**函式或函式庫**，讓其他程式呼叫使用。以 C 語言寫一個性能測試程式 (benchmark) 。
 - 功能一: 產生大量假資料 (十萬筆(以上) key-value) (註: 假資料格式自定義，key 需不同)，新增至作業一的 NoSQL DB 與 開源 Redis 的函式庫 (見 eeclass 參考資料 HiRedis)。測試延遲 (time complexity) 與使用的記憶體總量 (space complexity)，跟開源的 Redis 比較差距為何。
 - 功能二: 測試存取十萬筆(以上)平均延遲: 新增 (Create) 與 任意讀取 (Read) 分別量測。
 - 功能三: 分別觀察作業一 與 Redis 使用的記憶體容量，可以使用任何方法 (見 eeclass 參考資料)。
- **比較對象：**作業一 vs. Redis
- **步驟：**
 1. 產生假資料
 - 定義你的資料格式。例如，你的資料可以是一個 struct，包含 key 與 value。
 - 使用 C 語言函數生成十萬筆(以上)假資料。可用一個迴圈來做這件事，並在每一次迭代中生成一個新的 key-value pair。
 2. 測試性能
 - 寫一個函數計算可量測每次 key-value pair 對作業一與 Redis 資料庫“新增”的延遲。
 - 寫一個函數計算可量測每次 key-value pair 從作業一與 Redis 資料庫“讀取”的延遲。
 3. 計算延遲和空間複雜度
 - 計算出 平均寫入延遲 與 平均讀取延遲。
 - 顯示使用記憶體大小。
- **Deadline: 10/26 (將程式與文件報告上傳 github classroom)**