

計算機結構 Final Project Report

9721103 何書賢

流程：

1. 將資料所要的 4 個 cache 格式吃進程式，算出 K 值，利用 $K = \log_2 \frac{S}{L \times M}$ 可以得到 K 值。
2. 讀入 reference.lst。
3. 做第一個 Q_TABLE 判斷也就是 0 的數目和 1 的數目， $Q = \frac{\text{Min}(0 \text{ 的數目}, 1 \text{ 的數目})}{\text{Max}(0 \text{ 的數目}, 1 \text{ 的數目})}$ ，取最大值的項目，就是 D 的第一個 entry。
4. 做 C_table 也就是判斷第 i 項和第 j 項的相似程度，E 是一樣的個數，D 是不一樣的個數， $C = \frac{\text{Min}(E \text{ 的數目}, D \text{ 的數目})}{\text{Max}(E \text{ 的數目}, D \text{ 的數目})}$ 。
5. 將 Q 乘上上一個最佳化值對應到的 row，再比較大小，可以跑出 D 的第二個 entry。
6. 重複 step4、step5 可以做出一個 rank_table，就可以判斷 index 所要選取的是哪幾個。
7. 有了 rank_table 可以看 K 是幾個 entry 就可以找出 index 是哪幾位。
8. 利用 M、S、L、A 做出一個 TLB，另外設定一個特殊的 cycle counts 作為 LRU 的篩選使用，也就是將每一個指令做的時候就可以將 cycle counts 加 1，如此當越新的 instruction 其 cycle counts 會越大，反之越舊的就會越小，如此就可以做到 LRU 的目的。
9. TLB 的設計，一開始先切割好 TABLE 的格式，然後將 cycle counts 的值都設定成 -1，表示為沒有使用過，這樣就可以省去比對空的 Block。
10. 之後就是比較每一個 instruction 先將 instruction 的 index 算出，將 index 拿掉就是 tag 都可以存到 TLB 還有 cycle counts 也存進去。就可以比較 tag 是否正確，如果 miss 的話將該 index 中 cycle counts 小的值換掉。
11. 重複 9 的動作到結束。
12. 因為為了符合格式，特地將內容輸出到一個.txt，最後再一起整合。