**Computer Organization 2020**

**HOMEWORK 6**

系級: 資訊113 學號: F94081076 姓名: 郭立晨

**問題(Question)**

Q1. How do you know the number of block from input file?

Number of block=Cache size/Block size

Q2. How do you know how many set in this cache?

For Four-way: (Number of block /4)，(四個為一組)

For Direct-mapped: (Number of block /1)，(一個為一組)

For Fully: (Number of block/Number of block)=1，(全部只有一組)

Q3. How do you know the bits of the width of the Tag ?

Offset\_width=log2(Block\_size\*4)

Index\_width=log2(Number of block / (幾個一組))

Tag\_width(bits)=32bits – offset\_width – Index\_width

Q4. Briefly describe your data structure of your cache.

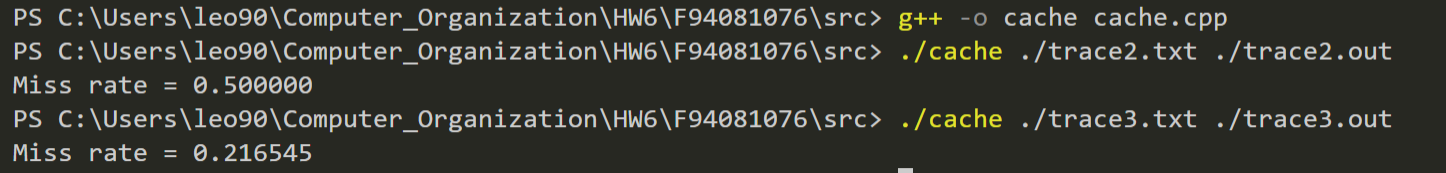
先宣告Cache Class，包含許多成員變數與函式，最重要的是代表cache的二維vector，此vector裡面的型態是pair<bool, unsigned>，每個element都包含判斷valid的布林變數以及儲存tag的unsigned變數。

Q5. Briefly describe your algorithm of LRU.

我將LRU分成兩個多載的函式，一個只用來更新cache中elements的順序，另一個用在當cache已經滿了要決定刪除哪一個element並插入新的tag。

首先，我將vector的頭視為較少使用的端點，尾巴則是較常使用的端點，也就是說，越新的且越常被使用的內容會靠近尾巴，反之則會靠近vector的頭。假設今天要刪除，則必定先刪除vector的頭，再將新的內容插入在vector的尾；若只是要調整順序，則是單純將要調整的element從原本的位置移到vector的尾，便代表此element已經更新成較常使用的element。

Q6. Run trace2.txt, trace3.txt and get the miss rate and put it in your report.



**心得(Report)**

(請寫下完成本次作業的心得、學到哪些東西、困難點的部分。)

(Please write your learned lesson and conclusion, and difficult point.)

這次的作業花費比較多心力，也比較難除錯，但是實作上課內容讓我覺得很充實，也更加了解上課內容的各種細節。我覺得最困難的就是實作LRU的部分，一開始還在想是不是要使用額外一個陣列來記住每個區塊的使用頻率，但是後來想到可以將vector中的排列順序作為使用頻率的紀錄依據，讓整個過程簡單許多，LRU也沒有花費幾行就寫出來了。整個作業做完很有成就感，也學到很多實作的技巧。