

# 國立臺灣科技大學答案卷

National Taiwan University of Science and Technology Answer Sheet

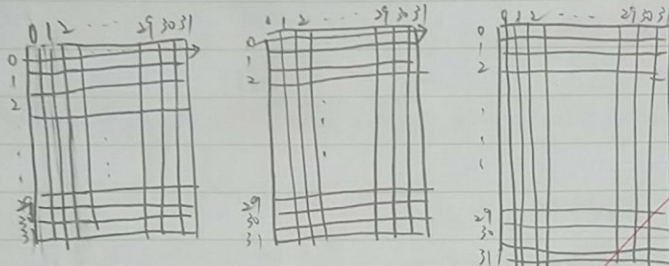
評分 Score	教師簽章 Signature of Lecturer
90	

姓名/Name 張恒豪 學號/Student ID B11002110 班級/Class 四電二乙

科目/Course title 計算機組織 日期/Date 111.12.23

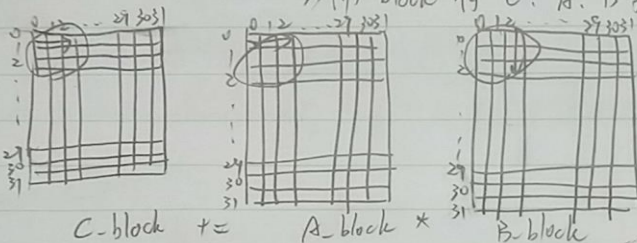
記分欄 從此處開始寫起。試卷用紙務須節用，非經主試認可不得續用其他紙張作答。/Please write from here.

1. 它是透過將要計算的矩陣拆分成多個小矩陣來做矩陣運算，例如  $32 \times 32$  矩陣相乘



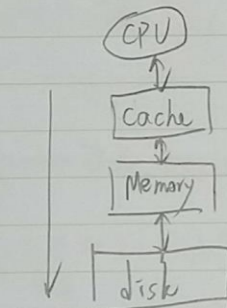
$C = A * B$

如果沒有拆成小矩陣，當要算 C 的某一列時如果 cache 的空間不夠塞下整個 B，就要再將 B 從 memory 再次載入。而 block 將 C、A、B 拆成多個小矩陣依序相乘則能避免此問題。



$C\text{-block} = A\text{-block} * B\text{-block}$

2. 我們利用 Memory Hierarchy 來完成這點。Memory Hierarchy 利用程式存取資料有 spatial 和 temporal locality 的特性來將儲存空間分成多個層級，例如：



越上層越小(越快)

越下層越大(越慢)

資料存取有連續性  
好存取一段時間  
不常存取  
一部分的記憶體

資料在短時間  
會再次被存取

注意：資料的流動只能在兩個相鄰的層級間進行

3. code: 0110\_0011\_1011

1	2	3	4	5	6	7	8	9	10	11	12
0	1	1	0	0	0	1	1	1	0	1	1
0011				0111				1001		1111 1100	

0011  
0111  
1001  
1011  
xor 1100  
1010

4. if (---)  
a = a + 10;  
if (---)  
a = a + 100;  
if (---)  
a = a + 1000;

write through:

當 a 有在 cache 時，每次 a 被加都要寫入 cache 再寫 memory。

write back:

當 a 有在 cache 時，每次 a 被加都只要寫入 cache

⇒ write back 能減少 PRAM access

pipe line

+5 因為要偵測是否有 hazard 需要知道每個 stage 會做哪些事，例如 beq 在 ID stage 算出結果還是 EX stage 才算出來

提前

addi x6, ~~x6~~<sup>x0</sup>, 08

outer loop:

addi x5, x0, 0

inner loop

addi x5, x5, 1

bne x5, x6, inner loop

如果用 1-bit predictor, 這個 misprediction 會發生在第  $n$  次執行到此行且  $x5=1$  時 ( $n > 1$ )

+8 因為 2-bit predictor 能消除這種 misprediction 的發生機會。

+5 因為 I-cache 的存取更具有 spatial locality 和 temporal locality。畢竟它通常一行一行讀取，而且很少大的 branch。

+5 8. L1 分成 I-cache 和 D-cache 是因為如果不這樣分會發生 structural hazard  
L2 用 higher set-associate 是因為要減少 miss rate。

+8 9. page table 是用來將 virtual address 轉成 physical address 的。TLB 是用來存放最近常用的 ~~TPE~~<sup>PTE</sup>，以減少 TLB 會用 higher-associative 或 fully-associative 是因為他要盡量減少由於使用 virtual memory 而額外增加的 memory 存取次數。

memory access 的次數

10. write through:

if write hit:  
寫 cache, 再寫 memory

if write miss:  
memory allocate, 再寫 memory

if read hit:  
直接讀 cache

if read miss:  
memory allocate, 再從 cache 讀

→ 假設全部 index 的 valid 皆為 1:  
直接把要更新的 block 覆蓋掉即可。

write back

if write hit:  
寫 cache

if write miss:  
memory allocate, 再寫 memory

if read hit:  
直接讀 cache

if read miss:  
memory allocate, 再從 cache 讀

→ 假設全部的 index 的 valid 皆為 1:  
檢查要替換掉的 block 有沒有 dirty, 如果有, 要把要替換掉的 block 的資料寫回 memory, 再將新進的 block 寫入

→ 因為發生 read miss 的隱對程序中有 memory allocate,

→ 而 write through 和 write back 的 memory allocate 動作不同, 所以 write through 和 write back 的機制會在 read miss 時被用到



# 國立臺灣科技大學答案卷

National Taiwan University of Science and Technology Answer Sheet

評分 Score	教師簽章 Signature of Lecturer

姓名/Name 張恒豪 學號/Student ID A11002110 班級/Class 四電二乙  
科目/Course title 計算機組織 日期/Date 11.12.23

記分欄 從此處開始寫起。試卷用紙務須節用，非經主試認可不得續用其他紙張作答。/Please write from here.

11. 不可能，因為如果一個 page 不存在在 Page Table 中，它就不會在 memory，而不在 memory 的 page 不可能會在 cache 中。

12. (5%) Please explain compulsory and capacity misses. Will conflict misses occur in a fully associative cache?

12. compulsory miss 是在開機時因為 cache 都是空的，所以一定會發生 cache miss。

capacity miss 是在有限的 cache 大小的情況下，cache miss 必有可能發生。

conflict miss 不可能發生在 fully associative cache。

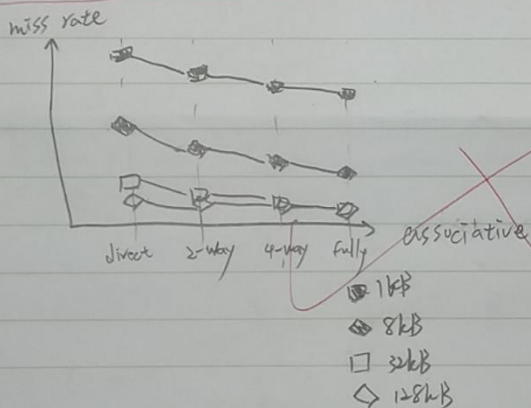
13. (a) 並沒有，可以用 write around 來減少剛開機時的 cache write miss

(b) 對，因為多個記憶體地址不會因為被編碼出的 index 相同而與彼此發生競爭。

(c) 對，可看 CH5 中的一張 simulation 圖表來得證！

13. (10%) Please comment on the following statements.

- There is no way to reduce compulsory misses.
- Fully associative caches have no conflict misses.
- In reducing misses, associativity is more important than capacity.



從此圖中可以看出當容量大到一定程度時，再繼續增加 cache 的容量(在此例中是 32KB 再增加成 128 KB，但都還是用 direct associativity)就無法有效地減少 miss rate。

而在此圖中也可以看出在 32KB 的 cache 上利用 4-way 的 associativity 能達到和用 128KB 2-way associativity 的 cache 有相當程度的 miss rate。

從上述兩段敘述可得證如果有效地減少 cache miss，cache associativity 會比 cache capacity 更重要。

14. 因為多個核心在存取到同一個 memory address，每個會複製同一份 memory address 的資料到同一個 cache index 中，此時如果用的 associative 的 way 數小於核心數，那同一個 cache index 中就沒辦法同時放進  $A_1, A_2, A_3$  和  $A_4$ ，因而造成 conflict miss。

address 的資料的副本分別為  $A_1, A_2, A_3$  和  $A_4$