

《计算机系统结构》作业 4

黄家晖 2014011330

一、计算 4 路组相联 Cache 运行程序时的缺失率。

1. 若 Cache 采用 Write-allocate 策略，计算缺失率

$i=0$ 时，假定 Cache 中没有数据，因此每读 4 个 int 就会有一次缺失和三次命中。这是因为采取了写分配策略，每次不命中都从主存中读取一个块（4 个 int）。经过计算，总共有 3072 次命中，1024 次缺失。

由于采用了四路组相联，如果策略使用得当，在 $i=0$ 循环完成之后，所有的 Cache 块应被填写了 M 数组的内容。因此当 $i=1$ 到 9 的时候，不存在 Cache 缺失。总共有 $9 \times 4096 = 36864$ 次命中。

因此，缺失率为 $M_1 = \frac{1024}{4096 \times 10} = 2.5\%$ 。

2. 若 Cache 采用 No-write allocate 策略，计算缺失率

简单考虑，在写不分配策略下，每次缺失不会将缺失的块调入主存，如果 Cache 初始为空，由于该程序中都是写操作，所以不会向 Cache 中填充任何内容。

因此，缺失率为 $M_2 = 100\%$

3. 针对 No-write allocate 策略，重新设计程序，使得缺失率最小

```
int M[4096], i, j, temp;
// Load M into cache.
for (i = 0; i < 4096; i++) {
    temp = M[i];
}
// Then all the loops below will hit.
for (i = 0; i < 10; i++) {
    for (j = 0; j < 4096; j++) {
        M[j] = i + j;
    }
}
```

新的程序设计如上所示，该程序首先通过一系列读操作将 M 数组装入 Cache，这样一来，之后的写操作均能命中。在这个程序运行的过程中，总共产生了 4096 次读访存和 40960 次写访存，其中读访存缺失 1024 次，写访存没有缺失。

新的缺失率为 $M'_2 = \frac{1024}{40960 + 4096} = 2.27\%$