5.16

6 因为使用中间位作为组索引可以使相邻的地址映射到同一组中，利用局部性原理提高缓存效率
高位作为标签可以在多级缓存系统中，将低层缓存中相同地址的块合并，提高缓存灵活性

7 这样做的好处在于：1°可以简化虚拟内存到物理内存的地址转换 2°可以用虚拟地址的页偏移位数来访问缓存，提高缓存的访问速度 3°使每个缓存块都可以存储一个完整的虚拟页面，提高缓存利用率。

8(1) 平均访问延时为 $T = 1\times 97\% + 110\times 3\% = 4.27$ 周期

(2) $1GB = 2^{20}KB$ ∴ 有 $\frac{2^0}{2^{20}} = 0.00006$ 的几率缓存命中

∴ 平均访问延时为 $T = 1\times 0.00006 + 110\times 0.99994 = 109.9934$ 周期

(3) 局部性原理是实际处理器能通过缓存极大减少存储系统平均访问延时的根本原因。因为时间空间的局部性原理，所以 cache 的命中率才可能达到 97%，否则只有 $\frac{1}{110}$

(4) 设命中率为 α $1\cdot\alpha + (110\cdot(1-\alpha)) \leq 105$ ∴ $5 < 109\alpha$ $\alpha > \frac{5}{109} \approx 0.0459$

9

地址位数 (bit)	缓存大小 (KB)	块大小 (Byte)	相联度	组数量	组索引位数	标签位数	偏移位数
32	4	64	2	32	5	21	6
32	4	64	8	8	3	23	6
32	4	64	全相联	1	0	26	6
32	16	64	1	256	8	18	6
32	16	128	2	64	6	19	7
32	64	64	4	256	8	18	6
32	64	64	16	64	6	20	6
32	64	128	16	32	5	20	7

10 (1) $0.22\cdot(1-p_1)+(100+0.22)p_1 < 0.52(1-p_2)+(100+0.52)p_2$

~~$99.78p_1 < 0.3+99.48p_2$~~ $100p_1 < 100p_2+0.3$, $p_1-p_2 < 0.003$

(2) $0.22(1-p_1)+0.22(k+1)p_1 < 0.52(1-p_2)+0.52(k+1)p_2$

$0.22kp_1-0.52kp_2 < 0.3$

11 0x为16进制 共16个块

直接映射时，有16个组 0x1001放入组1 0x1005放入组5 0x1021替换放入组1，0x1045、0x1305 0x2ee5 0x11a5依次替换组5 ∴共5次替换

2路组相联 8个组 0x1001放入组1 0x1005放入组5 0x1021放入组1 0x1045放入组5 后面3个请求都会发生替换 ∴共3次替换

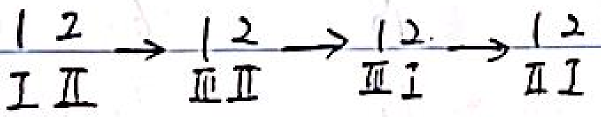
4路组相联 4个组 mod4 0x1001，0x1005，0x1021，0x1045都放入组1，后面3个请求也要放入组1 ∴3次替换

8路组相联 2个组 mod2 同理，所有数据都要放到组1，组1有8个位置，可以放下 ∴0次替换

12 块16Byte，总容量256Byte ⇒16个块 int32_t占4Byte 每块可放4个int32_t

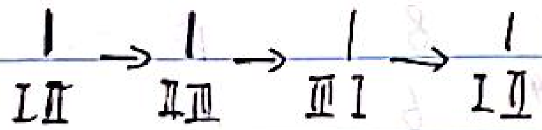
A: 8个组 每路可放 $8\times4=32$个int 共96个int，将其三等份，称为I，II，III

A中 第1路存I，第2路存II，存III时根据LRU，放在第1路。之后I存在第2路，…



N=100时，缺失率≈1/4

B: 16个组



缺失率≈1/4

13 改为

```
for(int j=0; j<128; ++j)
    for(int i=0; i<64; ++i)
        A[j][i]=A[j][i]+1;
```

14 块32字节 ⇒ 每块8个int　4KB中共128个块

(1) 优化前 直接映射　内存中 A[0][0]~A[0][63] 共占 $64 \times 4 = 2^8$ Byte

∴ $2^{12} \div 2^8 = 16$，~~每存放16个块~~ j (mod16) 时会存放在相同块

∵ j=0~127，i=0~64 ∴每次都缺失，次数为 $64 \times 128 = 8192$ 次

优化后 每块8个int ∴缺失率 $\frac{1}{8}$ ∴缺失次数 $64 \times 128 \div 8 =$ ~~8192~~ 1024次

(2) 全相联 优化前　缺失次数为 $64 \times 128 \div 8 = 1024$ 次

优化后　同上，为 $64 \times 128 \div 8 = 1024$ 次

(3) 优化前：要扩容到 $4KB \times \frac{128}{16} = 32KB$　优化后仍为4KB

15

	input 列0	列1	列2	列3	output 列0	列1	列2	列3
行0	miss	hit	miss	hit	miss	miss	miss	miss
行1	miss	hit	miss	hit	miss	miss	miss	miss
行2	miss	hit	miss	hit	miss	miss	miss	miss
行3	miss	hit	miss	hit	miss	miss	miss	miss

16 (1) 4个int/块　共32个块　2路组相联，共16组

缺失率 $\frac{1}{4}$，命中率 $\frac{3}{4}$

(2) 不能，因为增加缓存总大小无法减少强制缺失

(3) 可以，块变大可以减少强制缺失