

“计算机组织结构” 作业 5 参考答案

1. 计算机系统包含容量为 $32K \times 16$ 位的主存，按字编址，每字 16 位。Cache 采用 4 路组关联的映射方式，数据区大小为 4K 字，主存块大小为 64 字。假设 Cache 初始时是空的，处理器顺序地从存储单元（每个存储单元中包含 1 个字）0,1,...,4351 中取数，然后再重复这一顺序 9 次，并且 Cache 的速度是主存的 10 倍，同时假设块替换用 LRU 算法。请说明使用 Cache 后的速度为原来的多少倍（精度：小数点后 1 位）。

9.5

分析 1：主存地址为：标记 5，组号 4，块内地址 6

$4352/64=68$ ，即在前 68 块中操作 10 次，第一个轮回 68 次全部未命中，第二个轮回 0,1,2,3 四个组分别有编号未命中，未命中号为：0,1,2,3,16,17,18,19,32,33,34,35,48,49,50,51,64,65,66,67 共 20 次（因为只有四路，所以读取 64~67 的时候替换 0,1,2,3 那一组，下一次读取 0,1,2,3 的时候因为是 LRU 就去替换 16~19 那一组。以下类推，轮番替换，所以上面这 20 个数是永远未命中的）得命中率为 $P=(4352*10-68-20*9)/43520=99.43\%$

设 cache 的读取时间为 T，则主存的读取时间为 10T，则使用缓存后，系统效率提高到原来的 N 倍，N 为： $N=10T/(T+10*0.0057T)=9.5$

分析 2：也可参考紫皮书教材 P266。

2. 考虑一个每行 16 个字节的 4 行 Cache，主存按每块 16 个字节划分，即块 0 有地址 0 到 15 的 16 个字节，等等。先考虑以程序，它以如下地址顺序访问主存：

一次：63~70

循环 10 次：15~32，80~95

(2-1)假设 Cache 组织成直接映射式。块 0、4、...指派到行 0，块 1、5、...指派到行 1，如此类推。请计算命中率（形式：小数，非百分数；精度：小数点后 3 位）。

0.931

一次有 63, 64 未命中，循环第一次有 15, 16, 32, 80 未命中，以后 9 次有 16,80 未命中，所以命中率 $P=(8+18*10+16*10-2-4-2*9)/348=0.931$

[张鹤腾, 121250206]

(2-2)假设 Cache 组织成两路组关联映射式，共有两组，每组两行。偶序号块指派到组 0，奇序号块指派到组 1。使用 LRU 替换策略，请计算命中率（形式：小数，非百分数；精度：小数点后 3 位）。

0.983

前面一样，后 9 次循环都命中，所以 $P=(348-6)/348=0.983$

3. 考虑一个存取时间为 1ns 和命中率 $H=0.95$ 的 L1 Cache。假设我们修改了此 Cache 的设计（Cache 的容量、组织），从而使得命中率提升到 0.97，但也使存取时间增大到 1.5ns。如果要使得新设计能导致性能改善，cache 的速度必须是主存的多少倍以上（精度：整数）？

17

设主存的读写所耗为 T_{ns} ，必须满足：

$$T \cdot (1 - 0.95) + 1 > T \cdot (1 - 0.97) + 1.5 \quad \text{算得 } T > 25$$

所以 cache 的速度需要是主存的 $25/1.5=17$ 倍以上。

[杜铭哲, 181250028]

4. 假设某处理器的时钟频率为 1.2GHz, 当 L1 cache 无缺失时的 CPI 为 1 (即 CPU 可以快速地从 L1 cache 中读取指令, 并在 1 个时钟周期内完成)。访问一次主存的时间为 100ns (包括所有缺失处理), L1 cache 的局部缺失率为 2%。若增加一个 L2 cache, 并假定 L2 cache 的访问时间为 5ns, 而且其容量足够大到使全局缺失率仅为 0.5%。分析增加 L2 cache 后处理器执行程序的效率为原来的多少倍 (精度: 小数点后 3 位) ?

1.977

CPU 时钟周期为 $1/1.2\text{GHz}=0.833\text{ns}$

未增加 L2 时读一条指令平均耗时: $T_1=5/6+100 \cdot 2\%=2.833\text{ns}$

增加 L2 后: $T_2=5/6\text{ns}+2\% \cdot 5\text{ns}+100 \cdot 0.5\%\text{ns}=1.433\text{ns}$

则效率提高到了原来的 $T_1/T_2=1.977$ 倍。

5. 某计算机的主存地址空间为 256MB, 按字节编址, 指令 Cache 分离, 均有 8 个 Cache 行, 每个 Cache 行的大小为 64B, 数据 Cache 采用直接映射方式, 现有两个功能相同的程序 A 和 B, 其伪代码如下所示:

程序 A: <pre>int a[256][256]; int sum_array 1() { int i,j,sum=0; for(i=0;i<256;i++) for (j=0;j<256;j++) sum +=a[i][j]; return sum; }</pre>	程序 B: <pre>int a[256][256]; int sum_array 2() { int i,j,sum=0; for(j=0;j<256;j++) for (i=0;i<256;i++) sum +=a[i][j]; return sum; }</pre>
---	---

假定 int 类型数据用 32 位补码表示, 程序编译时 i、j、sum 均分配在寄存器中, 数组 a 的首地址为 320 (十进制)。

(6-1) 若不考虑用于 Cache 一致维护和替换算法的控制位, 则数据 Cache 的总容量为多少 (单位: 字节, 只填数字即可) ?

512

数据 Cache 有 8 个行, 每个行为 64B, 所以总容量为 $64\text{B} \cdot 8=512\text{B}$

(6-2) 数组元素 $a[0][31]$ 和 $a[1][1]$ 各自所在的主存块对应的 Cache 行号分别是多少 (Cache 行号从 0 开始) ? (答案以英文逗号分割)

6,5

内存大小为 256MB, 按字节寻址, 所以地址为 28 位; 每个块大小为 64B, 块内地址为 6 位; Cache 有 8 行, 行号为 3 位; 所以标记位为 $28-6-3=19$ 位。

数组 a 首地址为 320, 每个元素占 4 个字节, 所以 $a[0][31]$ 的地址为: $320+4 \cdot 31=444$ (十进制), 即

0...0110111100（二进制），所以行号为 110（二进制）即 6。

a[1][1]的地址为：320+（256*1+1）*4=1348（十进制），即 0...0 10101000100（二进制），所以行号为 101（二进制）即 5。

(6-3)程序 A 和 B 的数据访问命中率各是多少（形式：小数，非百分数,以英文逗号分割；精度:小数点后 3 位）？

0.938,0.000

数组 a 的大小为 $256*256*4B=218B$ ，占 $218B/64=212$ 个内存块，按行优先存放。程序 A 逐行访问数组 a，未命中次数未 212，所以命中率为 $(216-212)/216=0.938$ 。

程序 B 逐列访问数组 a，由于数组 a 一行的数据量 $256*4B=1KB>64B$ ，所以访问第 0 列时每个元素都不命中。由于数组 a 为 256 列，cache 仅有 8 行，当访问数组后续列时依然不命中。所以命中率为 0.000。

[庄子元, 181830266]

6. 【2009 统考真题】假设某计算机的存储系统由 Cache 和主存组成，某程序执行过程中访存 1000 次，其中访问 Cache 缺失(未命中) 50 次，则 Cache 的命中率是（ ）。

- A. 5%
- B. 9.5%
- C. 50%
- D. 95%

D

命中率 = Cache 命中次数/总访问次数。注意看清题目，题中说明的是缺失 50 次，而不是命中 50 次，仔细审题是做对题的第一步。

7. 【2009 统考真题】某计算机的 Cache 共有 16 块，采用二路组相联映射方式(即每组 2 块)。每个主存块大小为 32B，按字节编址，主存 129 号单元所在主存块应装入的 Cache 组号是（ ）。

- A. 0
- B. 2
- C. 4
- D. 6

C

由于 Cache 共有 16 块，采用二路组相联，因此共分为 8 组，组号为 0, 1, 2, ……，7。主存的某一字块按模 8 映射到 Cache 某组的任一字块中，即主存的第 0, 8, 16, ……字块可以映射到 Cache 第 0 组的任一字块中。每个主存块大小为 32B，因此 129 号单元位于第 4 块主存块中（注意是从 0 开始的），因此将映射到 Cache 第 4 组的任一字块中。

注意：由于在计算机系统结构和计算机组成原理的某些教材中介绍的组相联与此处的组相联不太相同，导致部分读者对题目理解错误。读者应以真题为准，以后再出现类似的题目，应以此种解答方式为标准。而且组号通常是从 0 而不是从 1 开始的（从选项也可看出）。

8. 【2012 统考真题】假设某计算机按字编址，Cache 有 4 行，Cache 和主存之间交换的块大小为 1 个字。若 Cache 的内容初始为空，采用二路组相联映射方式和 LRU 替换策略，则访问的主存地址

依次为 0,4,8,2,0,6,8,6,4,8 时，命中 Cache 的次数是()。

- A. 1
- B. 2
- C. 3
- D. 4

C

地址映射采用二路组相联，主存地址为 0~1、4~5、8~9 可映射到第 0 组 Cache 中，主存地址为 2~3、6~7 可映射到第 1 组 Cache 中。Cache 置换过程如下表所示。

走向		0	4	8	2	0	6	8	6	4	8
第 0 组	块 0		0	4	4	8	8	0	0	8	4
	块 1	<u>0</u>	<u>4</u>	<u>8</u>	8	<u>0</u>	0	<u>8*</u>	8	<u>4</u>	<u>8*</u>
第 1 组	块 2						2	2	2	2	2
	块 3				<u>2</u>	2	<u>6</u>	6	<u>6*</u>	6	6

注：“_”表示当前访问块，“*”表示本次访问命中。

注意：在不同的计算机组成原理教材中，关于组相联映射的介绍并不相同。通常采用上题中的方式，这也是唐朝飞所编教材中的方式，但本题中采用的是蒋本珊所编教材中的方式。可以推断两次命题的老师应该不是同一老师，因此给考生答题带来了困扰。

9. 【2014 统考真题】采用指令 Cache 与数据 Cache 分离的主要目的是 ()。

- A. 降低 Cache 的缺失损失
- B. 提高 Cache 的命中率
- C. 降低 CPU 平均访存时间
- D. 减少指令流水线资源冲突

D

把指令 Cache 与数据 Cache 分离后，取指和取数分别到不同的 Cache 中寻找，则指令流水线中取指部分和取数部分就可以很好地避免冲突,即减少了指令流水线的冲突。

10. 【2016 统考真题】有如下 C 语言程序段：

```
for(k=0; k<1000;k++)  
    a[k] = a[k] + 32;
```

若数组 a 和变量 k 均为 int 型，int 型数据占 4B，数据 Cache 采用直接映射方式，数据区大小为 1KB、块大小为 16B，该程序段执行前 Cache 为空，则该程序段执行过程中访问数组 a 的 Cache 缺失率约为 ()。

- A. 1.25%
- B. 2.5%
- C. 12.5%
- D. 25%

C

第一次访问 a[k]时，因为 Cache 中不存在，所以第一次 a[k]未命中，然后将该字的主存块调入 Cache 中。两次访问的 4 个整数，只有在第一次访问的第一个元素中缺失，其他 7 次都命中，所以结果为 1/8

=12.5%

11. 【2017 统考真题】某 C 语言程序段如下：

```
for(i=0;i<=9;i++){  
    temp=1;  
    for(j=0;j<=i;j++){  
        temp*=a[j];  
    }  
    sum += temp;  
}
```

下列关于数组 a 的访问局部性的描述中，正确的是（ ）。

- A. 时间局部性和空间局部性皆有
- B. 无时间局部性，有空间局部性
- C. 有时间局部性，无空间局部性
- D. 时间局部性和空间局部性皆无

A

时间局部性是，一旦一条指令执行，它就可能在不久的将来再被执行。空间局部性是，一旦一个存储单元被访问，它附近的存储单元也很快被访问。显然，这里的循环指令本身具有时间局部性，它对数组 a 的访问具有空间局部性，选 A。

12. 【2021 统考真题】若计算机主存地址为 32 位，按字节编址，Cache 数据区大小为 32KB,主存块大小为 32B，采用直接映射方式和回写（Write Back）策略，则 Cache 行的位数至少是（ ）。

- A. 275
- B. 274
- C. 258
- D. 257

A

Cache 数据区大小为 32KB，主存块的大小为 32B，那么 Cache 中共有 1K 个 Cache 行，物理地址中偏移量部分的长度为 5bit。因为采用直接映射方式，所以 1K 个 Cache 行映射到 1K 个分组，物理地址中组号部分的长度为 10bit。32bit 的主存地址除去 5bit 的偏移量和 10bit 的组号后，还剩 17bit 的 tag 部分。又因为 Cache 采用回写法，所以 Cache 行的总位数应为 32B（数据位）+17bit(tag 位）+1bit（脏位）+1bit（有效位）=275bit。

13. 【2015 统考真题】假定主存地址为 32 位，按字节编址，主存和 Cache 之间采用直接映射方式，主存块大小为 4 个字，每字 32 位，采用回写方式，则能存放 4K 字数据的 Cache 的总容量的位数至少是（ ）。

- A. 146K
- B. 147K
- C. 148K
- D. 158K

C

直接映射的地址结构为

主存字块标记	Cache 字块标记	字块内地址
--------	------------	-------

按字节编址，块大小为 $4 \times 32 \text{ 位} = 16\text{B} = 2^4\text{B}$ ，则“字块内地址”占 4 位；“能存放 4K 字数据的 Cache”即 Cache 的存储容量为 4K 字（注意单位），则 Cache 共有 $1\text{K} = 2^{10}$ 个 Cache 行，Cache 字块标记占 10 位；主存字块标记占 $32 - 10 - 4 = 18$ 位。

Cache 的总容量包括：存储容量和标记阵列容量（有效位、标记位、一致性维护位和替换算法控制位）。标记阵列中的有效位和标记位一定存在，而一致性维护位（脏位）和替换算法控制位的取舍标准是看词眼，题目中明确说明了采用回写法，则一定包含一致性维护位，而关于替换算法的词眼题目中未提及，所以不予考虑。因此，每个 Cache 行标记项包含 $18 + 1 + 1 = 20$ 位，标记阵列容量为 $2^{10} \times 20 \text{ 位} = 20\text{K 位}$ ，存储容量为 $4\text{K} \times 32 \text{ 位} = 128\text{K 位}$ ，总容量为 $128\text{K} + 20\text{K} = 148\text{K 位}$ 。

===== 分割线：以下内容不在小程序上提交 =====

14. 一个组关联 Cache 由 64 个行组成，每组 4 行。主存储器包含 4K 个块，每块 128 字，请表示主存地址的格式。

由每块 128 字得到块内地址长 7 位，64 行每组 4 行得一共 16 组，需要 4 位表示，标记需要 $12(4\text{K}) - 4(\text{组号}) = 8$ 位

标记	组号	块内地址
8	4	7

15. 一个两路组关联的 Cache 具有 8K 字节的容量，每行 16 字节。64M 字节的主存时字节可寻址的（即以字节为单位进行访问）。请给出主存地址格式。

根据每行 16 个字节，算出块内地址为 4；根据 2^9 行和 2 路组，算出组号为 8 位；根据主存为 64M 和每个块有 16 个字，算出有 2^{22} 个块，从而标记的位数为 $22 - 8 = 14$ 。即：

标记	组号	块内地址
14	8	4

[花霞, 121250049]

16. 假设 Cache 有 4K 字，每行 32 字。对十六进制主存地址：111111、666666、BBBBBB，请用十六进制格式表示如下信息：

- (1)直接映射 Cache 的地址格式，
- (2)全关联映射 Cache 的地址格式，
- (3)两路组关联 Cache 的地址格式。（提示：每个映射方式下，需要将标记、块内地址等分开表示。）

[刘璟, 121250083]

(1)共 6 位说明地址长 24，cache 一共有 $4\text{K}/32 = 2^7$ 行，即标记 12 行号 7 块内地址 5

标记(12 位)	行号(7 位)	块内地址(5 位)
111	08	11
666	33	06
BBB	5D	1B

(2)块号 19，块内地址 5

块号(19 位)	块内地址(5 位)
08888	11
33333	06
5DDDD	1B

(3)7 行两组表示，则组号 6。即标记 13，组号 6，块内地址 5

标记(13 位)	组号(6 位)	块内地址(5 位)
0222	08	11
0CCC	33	06
1777	1D	1B

17. 对一个有两级 Cache 的系统，定义：T_{C1} = 第一级 Cache 存取时间；T_{C2} = 第二级 Cache 存取时间；T_m=主存取时间为；H₁ = 第一级 Cache 命中率；H₂= 组合的第一/二级 Cache 命中率。请给出读操作时间的表示。

$$T_{\text{read}} = T_{C1} + (1 - H_1) * T_{C2} + (1 - H_2) * T_m$$

18. 假设主存中的 5 个块 {1,2,3,4,5} 映射到 cache 的同一组，对于主存块访问地址流 {1,2,3,4,1,2,5,1,2,3,4,5}，计算以下情况下的命中率（形式：小数，非百分数；精度：小数点后 3 位）：

（4-1）采用 3-路组关联和 LRU 算法

0.167

1	1	1	4	4	4	5	5	5	3	3	3
	2	2	2	1	1	1	1	1	1	4	4
		3	3	3	2	2	2	2	2	2	5

$$P = 2/12 = 1/6 = 0.167$$

（4-2）采用 4-路组关联和 LRU 算法

0.333

1	1	1	1	1	1	1	1	1	1	1	5
	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	5	5	5	5	4	4
			4	4	4	4	4	4	3	3	3

$$P = 4/12 = 1/3 = 0.333$$

（4-3）采用 5-路组关联和 LRU 算法

0.583

1	1	1	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	3	3	3	3	3	3
			4	4	4	4	4	4	4	4	4
						5	5	5	5	5	5

$$P = 7/12 = 0.583$$

(4-4) 采用 3-路组关联和 FIFO 算法

0.250

1	1	1	4	4	4	5	5	5	5	5	5
	2	2	2	1	1	1	1	1	3	3	3
		3	3	3	2	2	2	2	2	4	4

$$P=3/12=0.250$$

(4-5) 采用 4-路组关联和 FIFO 算法

0.167

1	1	1	1	1	1	5	5	5	5	4	4
	2	2	2	2	2	2	1	1	1	1	5
		3	3	3	3	3	3	2	2	2	2
			4	4	4	4	4	4	3	3	3

$$P=2/12=1/6=0.167$$

(4-6) 采用 5-路组关联和 FIFO 算法

0.583

1	1	1	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	3	3	3	3	3	3	3
			4	4	4	4	4	4	4	4	4
						5	5	5	5	5	5

$$P=7/12=0.583$$

19. 【2010 统考真题】其计算机的主存地址空间大小为 256MB，按字节编址。指令 Cache 和数据 Cache 分离，均有 8 个 Cache 行，每个 Cache 行大小为 64B，数据 Cache 采用直接映射方式。现有两个功能相同的程序 A 和 B，其伪代码如下所示：

程序 A:

```
int a[256][256];
int sum_array1()
{
    int i, j, sum = 0;
    for (i = 0; i < 256; i++)
        for (j = 0; j < 256; j++)
            sum += a[i][j];
    return sum;
}
```

程序 B:

```
int a[256][256];
int sum_array2()
{
```



```

int i, j, sum = 0;
for (j = 0; j < 256; j++)
    for (i = 0; i < 256; i++)
        sum += a[i][j];
return sum;
}

```

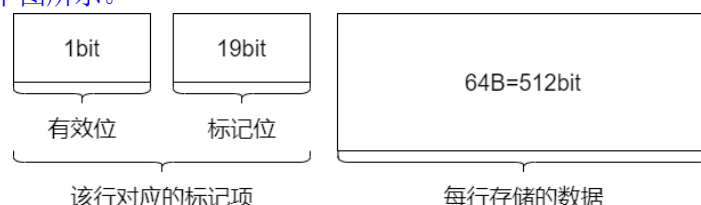
假定 int 类型数据用 32 位补码表示，程序编译时，i、j 和 sum 均分配在寄存器中，数组 a 按行优先方式存放，其首地址为 320（十进制数）。请回答下列问题，要求说明理由或给出计算过程。

1) 不考虑用于 Cache 一致性维护和替换算法的控制位，数据 Cache 的总容量为多少？

每个 Cache 行对应一个标记项，如下图所示：

有效位	脏位	替换控制位	标记位
-----	----	-------	-----

不考虑用于 Cache 一致性维护和替换算法的控制位。地址总长度为 28 位($2^{28} = 256\text{M}$)，块内地址 6 位($2^6=64$)，Cache 块号 3 位($2^3=8$)，因此 Tag 的位数为 $28-6-3=19$ 位，还需使用一个有效位，因此题中数据 Cache 行的结构如下图所示。



数据 Cache 共有 8 行，因此数据 Cache 的总容量为 $8 \times (64 + 20/8)\text{B} = 532\text{B}$ 。

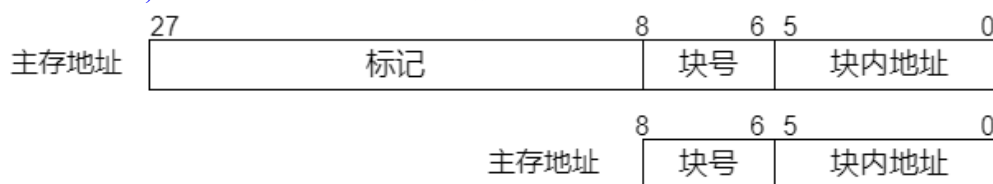
2) 数组元素 a[0][31]和 a[1][1]各自所在的主存块对应的 Cache 行号是多少(Cache 行号从 0 开始)？

数组 a 在主存的存放位置及其与 Cache 之间的映射关系如下图所示：

主存

数组按行优先方式存放，首地址为 320，数组元素占 4B。a[0][31]所在的主存块对应的 Cache 行号为 $[(320 + (0 \times 256 + 31) \times 4) \div 2^6] \bmod 2^3 = 6$ ；a[1][1]所在的主存块对应的 Cache 行号为 $[(320 + (1 \times 256 + 1) \times 4) \div 2^6] \bmod 2^3 = 5$ 。

【另解】由 1)可知主存和 Cache 的地址格式如下图所示。



数组按行优先方式存放，首地址为 320，数组元素占 4B。a[0][31]的地址为 $320 + 31 \times 4 = 110111100_{\text{B}}$ ，因此其对应的 Cache 行号为 $110_{\text{B}} = 6$ ；a[1][1]的地址为 $320 + 256 \times 4 + 1 \times 4 = 1348 = 10101000100_{\text{B}}$ ，因此其对应的 Cache 行号为 $101_{\text{B}} = 5$ 。

3) 程序 A 和 B 的数据访问命中率各是多少？哪个程序的执行时间更短？

数组 a 的大小为 $256 \times 256 \times 4\text{B} = 2^{18}\text{B}$ ，占用 $2^{13}/64 = 2^{12}$ 个主存块，按行优先存放，程序 A 逐行访问数组 a，共需访问的次数为 2^{16} 次，未命中次数为 2^{12} 次（即每个字块的第一个数未命中），因此程序 A 的命中率为 $(2^{16} - 2^{12}) / 2^{16} \times 100\% = 93.75\%$ 。

【另解】数组 a 按行存放，程序 A 按行存取。每个字块中存放 16 个 int 型数据，除访问的第一个不命中外，随后的 15 个全都命中，访问全部字块都符合这一规律，且数组大小为字块大小的整数倍，

因此程序 A 的命中率为 $15/16 = 93.75\%$ 。

程序 B 逐列访问数组 a，Cache 总数据容量为 $64B \times 8 = 512B$ ，数组 a 一行的大小为 1KB，正好是 Cache 容量的 2 倍，可知不同行的同一列数组元素使用的是同一个 Cache 单元，因此逐列访问每个数据时，都会将之前的字块置换出，即每次访问都不会命中，命中率为 0。由于从 Cache 读数据比从主存读数据快很多，所以程序 A 的执行比程序 B 快得多。

注意:本题考查 Cache 容量计算、直接映射方式的地址计算及命中率计算（行优先遍历与列优先遍历命中率差别很大）。

20. 【2020 统考真题】假定主存地址为 32 位，按字节编址，指令 Cache 和数据 Cache 与主存之间均采用 8 路组相联映射方式，直写(Write Through)写策略和 LRU 替换算法，主存块大小为 64B，数据区容量各为 32KB。开始时 Cache 均为空。请回答下列问题。

1) Cache 每一行中标记 (Tag)、LRU 位各占几位?是否有修改位?

主存块大小为 $64B = 2^6$ 字节，故主存地址低 6 位为块内地址，Cache 组数为 $32KB / (64B \times 8) = 64 = 2^6$ ，故主存地址中间 6 位为 Cache 组号，主存地址中高 $32 - 6 - 6 = 20$ 位为标记，采用 8 路组相联映射，故每行中的 LRU 位占 3 位，采用直写方式，故没有修改位。

2) 有如下 C 语言程序段:

```
for (k = 0; k < 1024; k++)  
    s[k] = 2 * s[k];
```

若数组 s 及其变量 k 均为 int 型，int 型数据占 4B，变量 k 分配在寄存器中，数组 s 在主存中的起始地址为 0080 00C0H，则在该程序段执行过程中，访问数组 s 的数据 Cache 缺失次数为多少?

$0080\ 00C0H = 0000\ 0000\ 1000\ 0000\ 0000\ 0000\ 1100\ 0000B$ ，主存地址的低 6 位为块内地址，为全 0，故 s 位于一个主存块的开始处，占 $1024 \times 4B / 64B = 64$ 个主存块；在执行程序段的过程中，每个主存块中的 $64B / 4B = 16$ 个数组元素依次读、写 1 次，因而对每个主存块，总是第一次访问缺失，此时会将整个主存块调入 Cache，之后每次都命中。综上，数组的数据 Cache 访问缺失次数为 64 次。

3) 若 CPU 最先开始的访问操作是读取主存单元 0001 0003H 中的指令，简要说明从 Cache 中访问该指令的过程，包括 Cache 缺失处理过程。

$0001\ 0003H = 0000\ 0000\ 0000\ 0001\ 0000\ 000000\ 000011B$ ，根据主存地址划分可知，组索引为 0，故该地址所在主存块被映射到指令 Cache 的第 0 组；因为 Cache 初始为空，所有 Cache 行的有效位均为 0，所以 Cache 访问缺失。此时，将该主存块取出后存入指令 Cache 的第 0 组的任意一行，并将主存地址高 20 位 (00010H) 填入该行标记字段，设置有效位，修改 LRU 位，最后根据块内地址 000011B 从该行中取出相应的内容。