6. 因为组索引越靠前的位数变化的范围越小，可以更好地利用局部性原理，提高缓存的命中率。而标签则需要存储的信息较多，需要更多位数来表示。

7. 使地址的组索引和块内偏移的总位数与虚拟内存系统的页偏移位数相同，可以使缓存系统的映射方式与虚拟内存系统的映射方式相同，方便管理和协调。

8.

1）存储系统平均访问延时 = 缓存命中时间 + 缓存缺失时间 × 缓存缺失率 = 1 + 110 × 3% + 105 × (1-3%) = 4.05 个周期。

2）每个块 64 字节，64KB 的 L1 缓存可以缓存 1024 个块，可以覆盖整个数组。由于完全随机访问，每个块的访问概率相等，因此平均访问延时等于缓存命中时间 + 缓存缺失时间 × 缓存缺失率 = 1 + 105 × 3% = 4.15 个周期。

3）局部性原理好坏直接影响了缓存的命中率和缺失率，进而影响处理器访存的性能。当程序的访问具有局部性时，缓存会取得更好的效果；当访问随机性很高时，缓存的效果就不如预期。

4）程序的平均缓存命中率需要高于 3%，才能让存储系统在使用 L1 时获得性能收益。

9.

| 编号 | 地址位数<br>Bit | 缓存大小<br>KB | 块大小<br>Byte | 相联度 | 组数量 | 组索引位数<br>Bit | 标签位数<br>Bit | 偏移位数<br>Bit |
|---|---|---|---|---|---|---|---|---|
| 1 | 32 | 4 | 64 | 2 | 32 | 5 | 27 | 5 |
| 2 | 32 | 4 | 64 | 8 | 8 | 3 | 29 | 5 |
| 3 | 32 | 4 | 64 | 全相联 | 1 | 0 | 32 | 5 |
| 4 | 32 | 16 | 64 | 1 | 256 | 8 | 24 | 5 |
| 5 | 32 | 16 | 128 | 2 | 64 | 6 | 30 | 6 |
| 6 | 32 | 64 | 64 | 4 | 256 | 8 | 28 | 5 |
| 7 | 32 | 64 | 64 | 16 | 64 | 6 | 26 | 5 |
| 8 | 32 | 64 | 128 | 16 | 32 | 5 | 27 | 6 |

10.

1）系统 A 的平均内存访问时间 = 缓存命中时间 + 缓存缺失时间 × 缓存缺失率 + 缓存缺失代价 × 缓存缺失率 = 0.22 + 100p1 + 110p2 > 0.52 + 100p2，即 p1 > 5p2 - 1.1。

2）系统 A 的平均内存访问时间 = 缓存命中时间 + 缓存缺失时间 × 缓存缺失率 + 缓存缺失代价 × 缓存缺失率 × k = 0.22 + 0.52k + 100p1k + 110p2k，而系统 B 的平均内存访问时间为 0.52 + 0.52k + 100p2k。令两者相等，可以得到 p1 > 1.75p2 - 0.3。

11.

直接映射，缓存容量为 16 × 64 ÷ 8 = 128 字节。

2 路组相联，缓存容量为 2 × 16 × 64 ÷ 8 = 256 字节，块替换次数为 1。

4 路组相联，缓存容量为 4 × 16 × 64 ÷ 8 = 512 字节，块替换次数为 0。

8 路组相联，缓存容量为 8 × 16 × 64 ÷ 8 = 1024 字节，块替换次数为 0。

12.
缓存 A 为 2 路组相联，总共可存放 8 个块，缓存 B 为直接映射，总共可存放 16 个块。
优化前，数组 A 在缓存 A 中的缺失次数为 640，缓存 B 中的缺失次数为 1600。
优化后，数组 A 在缓存 A 中的缺失次数为 128，缓存 B 中的缺失次数为 2510。

13.
将内层循环中访问 A 数组的方式修改为按照行优先，即将 A[j][i]修改为 A[i][j]，可以更好地利用行主元局部性，减少缓存缺失次数。

14.
1）优化前，总共发生 256 次缓存缺失；优化后，缓存缺失次数减少到 128 次，因为每次循环只需要缓存一行或一列（16 字节），因为块大小为 32 字节，所以每个块可以缓存 2 个元素，需要 256/2 = 128 个块。
2）优化前，总共发生 256 次缓存缺失；优化后，缓存缺失次数减少到 0 次，因为全相联缓存可以缓存所有数据，不会发生替换。
3）对于块大小 32 字节的直接映射缓存，每个块可以包含 2 个元素，缓存所需块数为 16 × 4 = 64 个。因此，优化前和优化后代码均需要至少 64 块的缓存容量。

15.

| | input 数组 | | | | output 数组 | | | |
|---|---|---|---|---|---|---|---|---|
| | 列 0 | 列 1 | 列 2 | 列 3 | 列 0 | 列 1 | 列 2 | 列 3 |
| 行 0 | miss | hit | hit | hit | miss | miss | miss | miss |
| 行 1 | Miss | hit | hit | hit | hit | hit | hit | hit |
| 行 2 | miss | hit | hit | hit | hit | hit | hit | hit |
| 行 3 | miss | hit | hit | hit | hit | hit | hit | hit |

16.
1）由于每次访问 input 数组时都会连续访问两个元素，因此每个缓存块中可以存储两个元素。因此，缓存中可以存储的块数为 512 / 16 / 2 = 16 个。而 input 数组的总大小为 2 x 128 x 4 = 1024 字节。因此，每个连续的访问块都有可能被替换出去，即每次访问都会引起缓存未命中。因此，缓存的命中率为 50%。

2)增加缓存的总大小不能改善该程序的命中率。因为该程序访问的 input 数组的大小为 1024 字节，而缓存的大小为 512 字节。无论缓存的大小增加到多少，都无法存储整个 input 数组，因此只要访问的是不在缓存中的数组元素，就会引起缓存未命中。

3）增加缓存的块大小可以改善该程序的命中率。由于每个块可以存储两个元素，因此块越大，每个元素被访问时可以引起的缓存命中次数就越多。缓存命中率得到改善。