Homework 12

1. Consider two different processors supporting 32-bit addresses with a *direct-mapped* cache design, with each bit designated as in the table. Answer the questions for each one.

	Tag	Index	Offset
P1	31-10	9-4	3-0
P2	31-12	11-5	4-0

A. How large is a block in each system? *(16 B; 32 B)*

B. How many blocks does the cache store? *(64 blocks; 128 blocks)*

C. How much data is stored in the cache? *(1 kB; 4 kB)*

D. How large does the cache have to be, including all overhead data? What percent of the cache is used for actual cached data? *(1216 B, 84.2%; 4448 B, 92.1%)*

2. A direct-mapped cache in a system with 32-bit addressing consists of slots for 64 different 16-byte blocks.

A. In the address, which bits are dedicated to the offset, index, and tag? *(tag: 10-31; index: 4-9, offset: 0-3)*

B. Starting with an empty cache, a program requests the following bytes:

12	20	36	8	1032	1056	24	32	2052	12	2088	1028

Make a table, giving for each request the block number, offset, index, tag, and whether the request is a hit or a miss. *(1028 is block 64, index 0, tag 1, miss; you're on your own for the others)*

C. What is the hit ratio of the above requests? *(16.67%)*

D. Which bytes are held in the cache after the last request? *(16-31, 1024-1039, 2080-2095)*

3. Repeat Problem 2, but this time using a set-associate cache of the same overall size, in which each of the 32 sets can hold 2 16-byte blocks. Use a *least-recently used* eviction strategy.

A. *(tag: 9-31; index: 4-8, offset: 0-3)*

B. *(1028 is block 64, index 0, tag 2, miss)*

C. *(25%)*

D. *(0-47, 1024-1039, 2080-2095)*

4. Consider two different caches with the properties as given in this table. Assume that a main-memory access takes 70 ns, and that 36% of all instructions are memory requests.

	Size	Miss Rate	Hit Time
C1	1 kB	11.4%	0.62 ns
C2	2 kB	8.0%	0.66 ns

A. If the clock cycle time of the CPU is set based on the time of a hit using each cache, what will the rate of the computer be? *(1.61 GHz; 1.52 GHz)*

B. What is the AMAT for each cache? How many cycles will memory access take on average? *(8.60 ns, 13.87 cyc; 6.26 ns, 9.48 cyc)*

C. Assuming all non-memory-access instructions take 1 cycle, what is the average time taken by one instruction? What is the average overall CPI? *(3.49 ns, 5.63 CPI; 2.68 ns, 4.05 CPI)*