ECE 30
Introduction to Computer Engineering

Example Problems, Set #9
Spring 2014

1. Cache C1 is direct-mapped with 16 one-word blocks. Cache C2 is direct-mapped with 4 four-word blocks. Assume that the miss penalty for C1 is 8 clock cycles and the miss penalty for C2 is 11 clock cycles. Assuming that the caches are initially empty,

   (a) find an example of a series of address references (given as word addresses) for which C2 has a lower miss rate but spends more memory bus clock cycles on cache misses than C1;

   (b) find an example of a series of address references (given as word addresses) for which C2 has more misses than C1.

2. Given is the following series of address references given as word addresses: 1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17. Assuming a direct-mapped cache with 16 one-word blocks that is initially empty, label each reference in the list as a hit or a miss and show the final contents of the cache (make sure to represent the cache structure correctly; show the index for every line of the cache and include valid and tag bits).

Address	Index	Tag	Hit/Miss
1			
4			
8			
5			
20			
17			
19			
56			
9			
11			
4			
43			
5			
6			
9			
17			

3. Using the series of references in Problem 2, show the hits and misses and final cache contents for a direct-mapped cache with four-word blocks and a *total size* of 16 words (make sure to represent the cache structure correctly; show the index for every line of the cache and include valid and tag bits as well besides the data).

Address	Block Offset	Index	Tag	Hit/Miss
1				
4				
8				
5				
20				
17				
19				
56				
9				
11				
4				
43				
5				
6				
9				
17				

4. Using the series of references given in Problem 2 and assuming a two-way set-associative cache (replacement scheme: least-recently-used) with two-word blocks and a *total size* of 16 words that is initially empty, label each reference in the table below as a hit or a miss and show the final contents of the cache (make sure to represent the cache structure correctly; show the index for every line of the cache and include all special bits as well besides the data).

Address	Block offset	Set index	Tag	Set entry	Hit/Miss
1					
4					
8					
5					
20					
17					
19					
56					
9					
11					
4					
43					
5					
6					
9					
17					

5. Using the series of references given in Problem 2 and assuming a fully associative cache (replacement scheme: least-recently-used) with four-word blocks and a *total size* of 16 words that is initially empty, label each reference in the table below as a hit or a miss and show the final contents of the cache (make sure to represent the cache structure correctly; show the index for every line of the cache and include all special bits as well besides the data).

Address	Block offset	Tag	Set Entry	Hit/Miss
1				
4				
8				
5				
20				
17				
19				
56				
9				
11				
4				
43				
5				
6				
9				
17				

6. Using the series of references given in Problem 2 and assuming a two-way set-associative cache (replacement scheme: least-recently-used) with four-word blocks and a *total size* of 16 words that is initially empty, label each reference in the table below as a hit or a miss and show the final contents of the cache (make sure to represent the cache structure correctly; show the index for every line of the cache and include all special bits as well besides the data).

Address	Block offset	Set index	Tag	Set entry	Hit/Miss
1					
4					
8					
5					
20					
17					
19					
56					
9					
11					
4					
43					
5					
6					
9					
17					