These examples came from Exercises 7.1, 7.2, and 7.3 in the text,  
Computer Organization and Design: The Hardware/Software Interface,  
by David Patterson and John Hennessy (from a previous edition).  
That text deals with word addresses.  
However, if you replace "word" by "byte" throughout this test file, it  
will not change the answer. These examples are ONLY to help you debug.  
The examples to hand in are in /course/csu380gc/hw8.  
  
For all problems, the cache holds a total of 16 words, the cache is initially  
empty and the address references (30-bit word addresses) are:  
  
1, 4, 8, 5, 20, 17, 19, 56, 9, 11, 4, 43, 5, 6, 9, 17  
  
We consider how the number of hits and misses changes as we consider different  
layouts of caches with a total of 16 words.  
  
7.1. 16 one-word blocks (cache entries), direct-mapped cache  
  
One-word entry, 2^0 = 1. So, the word offset within the entry requires 0 bits.  
There are 16 entries and 2^4 = 16. So, the 4 bits (bits 0-3) of the address  
represent the index.  
The remaining bits of the address (30 - 4 - 0) are the tag, which is 26 bits.  
  
addr tag index word\_offset hit/miss  
 1 0 1 0 miss  
 4 0 4 0 miss  
 8 0 8 0 miss  
 5 0 5 0 miss  
 20 1 4 0 miss (throw away data at address 4)  
 17 1 1 0 miss (throw away data at address 1)  
 19 1 3 0 miss  
 56 3 8 0 miss (throw away data at address 8)  
 9 0 9 0 miss  
 11 0 11 0 miss  
 4 0 4 0 miss (throw away data at address 4)  
 43 2 11 0 miss (throw away data at address 11)  
 5 0 5 0 hit  
 6 0 6 0 miss  
 9 0 9 0 hit  
 17 1 1 0 hit  
  
Final state of cache:  
 valid tag data(unspecified, 1 word)  
 index # +------+-------+----------------------------+  
 0 | 0 | X | |  
 1 | 1 | 1 | |  
 2 | 0 | X | |  
 3 | 1 | 1 | |  
 4 | 1 | 0 | |  
 5 | 1 | 0 | |  
 6 | 1 | 0 | |  
 7 | 0 | X | |  
 8 | 1 | 3 | |  
 9 | 1 | 0 | |  
 10 | 0 | X | |  
 11 | 1 | 2 | |  
 12 | 0 | X | |  
 13 | 0 | X | |  
 14 | 0 | X | |  
 15 | 0 | X | |  
 +------+-------+----------------------------+  
7.2. 4 four-word blocks (cache entries), direct-mapped cache  
  
Four-word entry, 2^2 = 4. So, the word offset within the entry requires  
2 bits. The 2 bits (0-1) of the address represent the word within the entry.  
There are 4 entries and 2^2 = 4. So, the 2 bits (bits 2-3) of the address  
represent the index.  
The remaining bits of the address (30 - 2 - 2) are the tag, which is 26 bits.  
  
addr tag index word\_offset hit/miss  
 1 0 0 1 miss  
 4 0 1 0 miss  
 8 0 2 0 miss  
 5 0 1 1 hit  
 20 1 1 0 miss (throw away data at address 5)  
 17 1 0 1 miss  
 19 1 0 0 hit  
 56 3 2 0 miss (throw away data at address 8)  
 9 0 2 1 miss (throw away data at address 56)  
 11 0 2 3 hit  
 4 0 1 0 miss (throw away data at address 20)  
 43 2 2 3 miss (throw away data at address 11)  
 5 0 1 1 hit  
 6 0 1 2 hit  
 9 0 2 1 miss (throw away data at address 43)  
 17 1 0 1 hit  
  
Final state of cache:  
 valid tag data(unspecified, 4 words)  
 index # +------+-------+----------------------------+  
 0 | 1 | 1 | |  
 1 | 1 | 0 | |  
 2 | 1 | 0 | |  
 3 | 0 | X | |  
 +------+-------+----------------------------+  
  
7.3. 16 one-word blocks (cache entries), two-way set associative  
Assume LRU replacement. (This means that if there are two valid cache entries  
for a given index, and if we have to throw away one to make room, then we  
throw away the one that was Least Recently Used.)  
  
One-word entry, 2^0 = 1. So, the word offset within the entry requires 0 bits.  
There are 16 entries, but it is two-way set associative. So, we have only 8  
different indices (indices 0-7). Next 2^3 = 8. So, the 3 bits (bits 0-2) of  
the address represent the index.  
The remaining bits of the address (30 - 3 - 0) are the tag, which is 27 bits.  
  
addr tag index word\_offset hit/miss  
 1 0 1 0 miss  
 4 0 4 0 miss  
 8 1 0 0 miss  
 5 0 5 0 miss  
 20 2 4 0 miss (2 valid entries: 4, 20)  
 17 2 1 0 miss (2 valid entries: 1,17)  
 19 2 3 0 miss  
 56 7 0 0 miss (2 valid entries: 8, 56)  
 9 1 1 0 miss (2 valid entries: 9, 17; remove data, addr 1)  
 11 1 3 0 miss  
 4 0 4 0 hit (2 valid entries: 11, 19)  
 43 5 3 0 miss (2 valid entries: 43, 11; rem. data, addr 19)  
 5 0 5 0 hit  
 6 0 6 0 miss  
 9 1 1 0 hit (2 valid entries: 9, 17)  
 17 2 1 0 hit (2 valid entries: 9, 17)  
  
Final state of cache:  
 valid tag data(unspecified, 1 word)  
 index # +------+-------+----------------------------+  
 0 | 1 | 1 | |  
 0 | 1 | 7 | |  
 1 | 1 | 1 | |  
 1 | 1 | 2 | |  
 2 | 0 | X | |  
 2 | 0 | X | |  
 3 | 1 | 5 | |  
 3 | 1 | 1 | |  
 4 | 1 | 0 | |  
 4 | 1 | 2 | |  
 5 | 1 | 0 | |  
 5 | 0 | X | |  
 6 | 1 | 0 | |  
 6 | 0 | X | |  
 7 | 0 | X | |  
 7 | 0 | X | |  
 +------+-------+----------------------------+