7.11 The miss penalty is the time to transfer one block from main memory to the cache. Assume that it takes one clock cycle to send the address to the main memory.

a. Configuration (a) requires 16 main memory accesses to retrieve a cache block and words of the block are transferred 1 at a time.

   Miss penalty = $1 + 16 \times 10 + 16 \times 1 = 177$ clock cycles.

b. Configuration (b) requires 4 main memory accesses to retrieve a cache block and words of the block are transferred 4 at a time.

   Miss penalty = $1 + 4 \times 10 + 4 \times 1 = 45$ clock cycles.

c. Configuration (c) requires 4 main memory accesses to retrieve a cache block and words of the block are transferred 1 at a time.

   Miss penalty = $1 + 4 \times 10 + 16 \times 1 = 57$ clock cycles.

7.32 The total size is equal to the number of entries times the size of each entry. The number of entries is equal to the number of pages in the virtual address, which is

$$\frac{2^{40}\ \text{bytes}}{16\ \text{KB}} = \frac{2^{40}\ \text{bytes}}{2^4\ 2^{10}\ \text{bytes}} = 2^{26}$$

The width of each entry is 4 + 3 bits = 40 bits = 8 bytes. Thus the page table contains 2^{29} bytes or 512 MB!

8.3 After reading sector 7, a seek is necessary to get to the track with sector 8 on it. This will take some time (on the order of a millisecond, typically), during which the disk will continue to revolve under the head assembly. Thus, in the version where sector 8 is in the same angular position as sector 0, sector 8 will have already revolved past the head by the time the seek is completed and some large fraction of an additional revolution time will be needed to wait for it to come back again. By skewing the sectors so that sector 8 starts later on the second track, the seek will have time to complete, and then the sector will soon thereafter appear under the head without the additional revolution.

8.8 We determine an average disk access time of 8 ms+ 4.2 ms + .2 ms + 2 ms = 14.4 ms. Since each block processed involves two accesses (read and write), the disk component of the time is 28.8 ms per block processed. The (non-overlapped) computation takes 20 million cycles at 400 MHz, or another 50 ms. Thus, the total time to process one block is 78.8 ms, and the number of blocks processed per second is simply 1 / 0.0788 = 12.7.

NT-1. Consider a 4-way set associative cache that stores 2048 sets of 4-word blocks. This works with a byte-addressable DRAM with 32-bit addresses and 32-bit words. Specify exactly what kind of information is in the cache at each addressable location, taking into account whether the write policy is write back or write thru.

Each location (set) contains 4 blocks. Each block consists of a valid bit, a tag, 4 32-bit data words, and, if the write policy is write-back, a dirty bit. Since the cache has 2048 (= 2^11) sets or addressable locations, 11 bits of the address word are used to access the cache. These are bits 14-4 (bits 3, 2 locate a word in the block, 1 and 0 are byte locations). Bits 31-15 of the address make up the tag, i.e., there is a 17-bit tag in each block. So, for a write-thru cache, each block in the set contains 1+1+17+4x32= 147 bits, and each set contains 4x147=588 bits for a write-back cache, or 584 bits for a write-thru cache.

NT-2. If the word at the following DRAM address is in the cache specified in the previous question, where would it be and how would it be found? 1010 1111 0000 0011 1100 0011 1111 1100

The cache is addressed by bits 14-4, which are 100 0011 1111. After reading out the set at that location, we match the tag: 1010 1111 0000 0011 1 against the tags for the 4 blocks in the set, also matching the valid bits against 1. If the word is in the cache, there will be a match and a valid bit equal to 1. We then, within the block, use the MUX to select the word from the block at the internal location given by the word location bits, 01 (bits 3 and 2) of the DRAM address.

NT-3. Key parameters of a magnetic disk drive are the seek time, rotational latency, controller time, transfer rate, and disk capacity.
(a) Suppose a technology improvement allowed us to double the number of bits on a track, without changing the disk diameter or the number of tracks. How, if at all, would that affect each of the above listed parameters?
(b) Suppose, instead, the number of tracks could be doubled, without changing the disk diameter or the number of bits per track. How would this affect each of the key parameters?

(a) The transfer rate and the disk capacity would both be doubled, since we obviously are packing twice as many bits into the disk and, in a given unit of time, twice as many bits pass under the head for reading or writing. The other parameters are not affected.

(b) Only the disk capacity would be changed; it would be doubled. (The assumption here is that the number of tracks per inch of radius is doubled, without changing the region on the disk that is occupied by the tracks. If the tracks extended over a greater part of the disk radius, then the seek time would be increased.)