Exercise Number: 6.2.6

In this problem we are required to fill in the given table related to cache memories/specs. It is quite simple once the relation $C = S \times B \times E$ is recalled, as well as $t = m - s - b$.

Cache	m	C	B	E	S	t	s	b
1	32	2048	8	1	256	21	8	3
2	32	2048	4	4	128	23	7	2
3	32	1024	2	8	64	25	6	1
4	32	1024	32	2	16	23	4	5

Discussion.

In the previous problem I wrote:

"Also in general there are quite a few tag bits. The larger the number of tag bits, the greater the disparity between the size of the cache and the size of main memory (or whatever the lower level is that it's calling from)."

However, I realized I probably erred in writing thusly. Take a look at caches 1 and 2 in the above problem. Both are 2048 bytes, or encoded using 11 bits. Main memory, on the other hand, is 32 bits in both cases, and so the relative difference between the cache size and memory size is the same in both cases.

Looking at it more closely, the tag bits are needed to differentiate data *within* set slots. So really, the larger the number of tag bits, the larger number of data slots are needed to differentiate between potential memory addresses in a single set. That can be caused by several factors, including block size, the number of sets, *and* the size of memory (notice: NOT E). In all cases, if eveverything else is kept constant,

1. Increasing the number of sets decreases tag bits, since we dont have as many memory locations per set.
2. Increasing the size of blocks again decreases the number of tag bits, since we can support more unambiguous memory per line.
3. Increasing the size of the lower cache bits, or m, increases tag bits required obviously since we have more memory ot keep track of for fixed block size and number of sets.

In other words, $m = s + t + b$, as defined. E and t are more or less entirely unrelated, they areboth determined by fixed memory, cache size, number of sets and block size.