Name: _____

Directions: Work only on this sheet (on both sides, if needed); do not turn in any supplementary sheets of paper. There is actually plenty of room for your answers, as long as you organize yourself BEFORE starting writing. In order to get full credit, SHOW YOUR WORK.

1. (5) The more-significant byte of Word 1 in Fig.16.10 is in Chip #_____.

2. (5) The VPN field in inverted page tables is analogous to the _____ field in _____ caches.

3. (5) A TLB is a subset of a _____.

4. (5) OSs use a write-_____ policy for disk caches.

5. Look at Fig. 17.13 and Table 17.4 in Dandamudi.

(a) (10) Suppose the block access pattern in the table were to change to 0,1,6,7,11,12. Show the new table that would result.

(b) (5) Re-do (a), but for the situation in which the cache size is 32 bytes instead of 16. (The set size is still 2.)

(c) (5) In the original setup (i.e. 16-byte cache), Byte 2 of Block 11 has address _____.

6. (10) In an entry in a Pentium page table, if Bit 0 is 0, then what information is likely stored in the rest of the entry? Be specific.

7. In Fig.16.11, suppose we want to construct a 128Mx8 system, still using 16Mx16 chips as our building blocks. As in the original 64Mx32 system, we still want consecutive words to be in the same module (except, as usual, at boundaries). We will have a MUX between the data bus and the CPU. We will not waste space.

(a) (10) We will need _____ chip(s) per row, and _____ rows.

(b) (10) Lines _____ from the address bus will feed into a ____-to-_____ decoder.

(c) (5) Each module connects to lines _____ of the data bus.

8. (10) On the Pentium, software can make certain regions of memory cacheable/noncacheable. The smallest possible size of such a region is _____ blocks.

9. In an nxn matrix A with elements a_{ij}, the k^{th} *subdiagonal* is defined to consist of all the elements $a_{r+k,r}$. So for example if A is 6x6, its second subdiagonal consists of a_{20}, a_{31}, a_{42}, and a_{53}. (Row and column numbering start at 0.) Suppose we have a program which traverses the k^{th} subdiagonal of A.

(a) (10) What will be the stride? Your answer must be an expression in n and k.

(b) (5) Say our memory consists of n modules, low-order interleaved (ordinary, not skewed). How many modules can we keep busy at once? Assume that a_{00} starts in the first (i.e. lowest-address) module. Answer for general n if possible; otherwise, answer for n = 6.

Solutions:

1. 3

2. tag, associative (not just set- or fully-associative)

3. page table (also accepted CPU as an answer)

4. back

5.a.

block	h/m	s0,l0	s0,l1	s1,l0	s1,l1
0	m	0	-	-	-
1	m	0	-	1	-
6	m	0	6	1	-
7	m	0	6	1	7
11	m	0	6	11	7
12	m	12	6	11	7

5.b.

Each set is still 2 lines, and each line is still 4 bytes, so each set is still 8 bytes. Thus there will be 4 sets for this 32-byte cache. The lowest 2 bits of the block number will determine the set number. E.g. block $6 = 0110_2$ will be in set 2.

block	h/m	s0,l0	s0,l1	s1,l0	s1,l1	s2,l0	s2,l1	s3,l0	s3,l1
0	m	0	-	-	-	-	-	-	-
1	m	0	-	1	-	-	-	-	-
6	m	0	-	1	-	6	-	-	-
7	m	0	-	1	-	6	-	7	-
11	m	0	-	1	-	6	-	7	11
12	m	0	12	1	-	6	-	7	11

5.c.

Since there are 4 bytes per block, the block number for address A is $\lfloor A/4 \rfloor$, and A's byte number within the block is A mod 4. So, Byte 2 in Block 11 corresponds to address $11 \times 4 + 2 = 46$.

6. The most important information stored here would be the page's location on disk. Bits 7, 4 and 3 would also likely be preserved.

7. Each chip word holds 16 bits, which is 2 system words. So, system addresses 0-32M will be stored in the first 16Mx16 chip, system addresses 32M-64M will be stored in the second 16Mx16 chip, etc. From this you can see that we need 4 rows, 1 chip per row. Bus lines A25-A26 will determine the row. The chip will then supply (in the case of a read) a 16-bit chip word on bus lines D0-D15. A0 will then determine whether we access the high byte or the low byte of the 16-bit chip word which is accessed.

8. Cacheability in the Pentium can only be controlled at the page level. The page size is 2^{12} bytes, and the block size is 2^5 bytes, so a page is $2^7 = 128$ blocks.

9.a. Each row contains n elements, so two consecutive elements in the same column are n words apart in memory. But consecutive elements in a subdiagonal are not in the same column; the second one is one column to the right of the first, so they are n+1 words apart. Thus the stride is n+1.

9.b. Since n and n+1 are relatively prime, there will be no memory module conflicts. Thus all n-k subdiagonal elements can be accessed at once. (Note: Credit was not normally given for part (b) for those who missed part (a), unless convincing justification was given in (b).)