COMP 222
Sample Midterm #2

Problem 1 (15 points)

Suppose you receive the Hamming code shown below containing both data bits and even-parity check bits (bits are numbered 24 down to 1, left to right, in other words, the leftmost bit is code bit 24 and the rightmost bit is code bit 1).

a) Assuming there is at most a 1-bit error, using Hamming's algorithm, determine which code bit position has an incorrect value, and what its value should be.

Hamming code: **0110 1011 0110 0001 1101 0110**

$P_1 = 0$ $P_2 = 1$ $P_4 = 0$ $P_8 = 1$ $P_{16} = 0$

Denoting code bit ***i*** as C_i:

$P_1' = C_3 \oplus C_5 \oplus C_7 \oplus C_9 \oplus C_{11} \oplus C_{13} \oplus C_{15} \oplus C_{17} \oplus C_{19} \oplus C_{21} \oplus C_{23} = 1$ (7 1's)
$P_2' = C_3 \oplus C_6 \oplus C_7 \oplus C_{10} \oplus C_{11} \oplus C_{14} \oplus C_{15} \oplus C_{18} \oplus C_{19} \oplus C_{22} \oplus C_{23} = 1$ (7 1's)
$P_4' = C_5 \oplus C_6 \oplus C_7 \oplus C_{12} \oplus C_{13} \oplus C_{14} \oplus C_{15} \oplus C_{20} \oplus C_{21} \oplus C_{22} \oplus C_{23} = 1$ (7 1's)
$P_8' = C_9 \oplus C_{10} \oplus C_{11} \oplus C_{12} \oplus C_{13} \oplus C_{14} \oplus C_{15} \oplus C_{24} = 1$ (3 1's)
$P_{16}' = C_{16} \oplus C_{17} \oplus C_{18} \oplus C_{19} \oplus C_{20} \oplus C_{21} \oplus C_{22} \oplus C_{23} \oplus C_{24} = 1$ (5 1's)

	P_{16}	P_8	P_4	P_2	P_1	
$\oplus$	P_{16}'	P_8'	P_4'	P_2'	P_1'	
	1	0	1	0	1	$\rightarrow$ Bit 21 is in error; it should be 1

b) Suppose the above detected error is actually the result of two erroneous non-parity code bit positions. Give a possible combination of two erroneous non-parity code bit positions that would result in the above detected one erroneous code bit.

Any two bits whose bit-wise XOR is 10101, such as 00011 (3) and 10110 (22)

c) Suppose the above detected error is actually the result of three erroneous non-parity code bit positions. Give a possible combination of three erroneous non-parity code bit positions that would result in the above detected one erroneous code bit.

Any three bits whose bit-wise XOR is 10101, such as 00111 (7), 00110 (6), 10100 (20)

Problem 2 (10 points)

Suppose we want to compare two disks in terms of average access time. Disk #1 has an average seek time of 7.4ms, a rotational delay of 5400rpm, a track size of 800 sectors/track, and a sector size of 512 bytes/sector. Disk #2 has an average seek time of 9.5ms, a rotational delay of 7200rpm, a track size of 600 sectors/track, and a sector size of 512 bytes/sector.

(a) Assuming a total access of 1.2288MB (1MB=10^6 bytes) and random access, determine the average access time for both Disk #1 and Disk #2.

Disk #1: T_s = 7.4ms r=5400rpm b=512 bytes/sector N=512*800=409600 bytes

Time per sector: $T = T_s + (1/2)(1/r) + (1/r)(b/N)$ = 12.97ms

Total sectors to read: (1.2288MB)/(512 bytes/sector) = 2400 sectors

Total time = 2400 * 12.97ms = 31.13 sec.

Disk #2: T_s = 9.5ms r=7200rpm b=512 bytes/sector N=512*600=307200

Time per sector: $T = T_s + (1/2)(1/r) + (1/r)(b/N)$ = 13.68ms

Total sectors to read: (1.2288MB)/(512 bytes/sector) = 2400 sectors

Total time = 2400 * 13.68ms = 32.83 sec.

(b) Assuming a total access of 1.2288MB (1MB=10^6 bytes) and sequential access, determine the average access time for both Disk #1 and Disk #2.

Disk #1: T_s = 7.4ms r=5400rpm b=N =512*800 bytes/track

Number of tracks: (1.2288MB) / (512 bytes/sector * 800 sectors/track) = 3 tracks
Time for 1st track: $T = T_s + (1/2)(1/r) + (1/r)(b/N)$ = 24.06ms
Time for other 2 tracks: $(1/2)(1/r) + (1/r)(b/N)$ = 16.66ms
Total time = 24.06 + 2(16.66) = 57.38 ms

Disk #2: T_s = 9.5ms r=7200rpm b=N=512*600 bytes/track

Number of tracks: (1.2288MB) / (512 bytes/sector * 600 sectors/track) = 4 tracks
Time for 1st track: $T = T_s + (1/2)(1/r) + (1/r)(b/N)$ = 22ms
Time for other 3 tracks: $(1/2)(1/r) + (1/r)(b/N)$ = 12.5ms
Total time = 22 + 3(12.5) = 59.5 ms

Problem 3 (10 points)

A byte-addressable virtual memory has a page size of 1024 bytes. The byte-addressable physical memory also has a page size of 1024 bytes and 4 physical page frames. Initially the physical memory is empty. Virtual pages will be mapped to physical pages in the order: 0,1,2,3 whenever a page fault occurs until the physical memory is full. The protocol for further page faults uses the least-recently used (LRU) scheme. Determine what physical addresses (PA) are returned for the following virtual addresses (VA) by filling in the table below. If the reference causes a fault, state so.

VA	2047	4096	4095	2049	3071	3073	5108	1025	1023	512
PA	**Fault**	**Fault**	**Fault**	**Fault**	4095	2049	2036	1	**Fault**	3584

2047: VP 1 → Fault → ___, ___, ___, _1_
0

4096: VP 4 → Fault → ___, ___, _1_, _4_
0 1

4095: VP 3 → Fault → ___, _1_, _4_, _3_
0 1 2

2049: VP 2 → Fault → _1_, _4_, _3_, _2_
0 1 2 3

3071: VP 2 → PA=3(1024) + (3071-2(1024)) = 4095 → _1_, _4_, _3_, _2_
0 1 2 3

3073: VP 3 → PA=2(1024) + (3073-3(1024)) = 2049 → _1_, _4_, _2_, _3_
0 1 3 2

5108: VP 4 → PA=1(1024) + (5108-4(1024)) = 2036 → _1_, _2_, _3_, _4_
0 3 2 1

1025: VP 1 → PA=0(1024) + (1025-1(1024)) = 1 → _2_, _3_, _4_, _1_
3 2 1 0

1023: VP 0 → Fault → _3_, _4_, _1_, _0_
2 1 0 3

512: VP 0 → PA=3(1024) + (512-0(1024)) = 3584 → _3_, _4_, _1_, _0_
2 1 0 3

Problem 4 (15 points)

a) Determine the IEEE-754 single-precision representation for the decimal number: -3.8671875**.** Write the answer as eight hexadecimal digits.

Step performed	Result
Decimal → Binary	-11. 1101111
Normalize	$-1.11101111 * 2^{1}$
Bias exponent	$1+127 = 128 = (10000000)_2$
Append sign, exponent, mantissa	1 10000000 111011110…0
Binary → HEX	C0778000

b) Determine the decimal representation for the floating-point number represented in IEEE-754 single-precision format as: 4497E000**.** Write the result without any exponent as a decimal number.

Step performed	Result
HEX → Binary	0100 0100 1001 0111 1110 0…0
Extract sign, exponent, mantissa	0 10001001 0010111 1110 …0
Unbias exponent	10001001 → 137-127=10
Unnormalize	$1.0010111111 * 2^{10} = 10010111111$
Binary → Decimal	10010111111 → 1215