Spring 2009 CS 320 Exam 3 Review

Note: Keep in mind that this review is designed as a practice exam & is not intended to be comprehensive. While studying for the exam, it would be prudent to consider variations on the questions presented here. You should also include the homework questions when reviewing for the exam. Finally, it is possible to get strong clues for answering some of the questions in the review by referring to other questions. In general, you should not depend on in the actual exam.

1. Consider the following Boolean function: F = AB + (AC)'
a. Write out the truth table for the function (note you may have more grid than you really need)

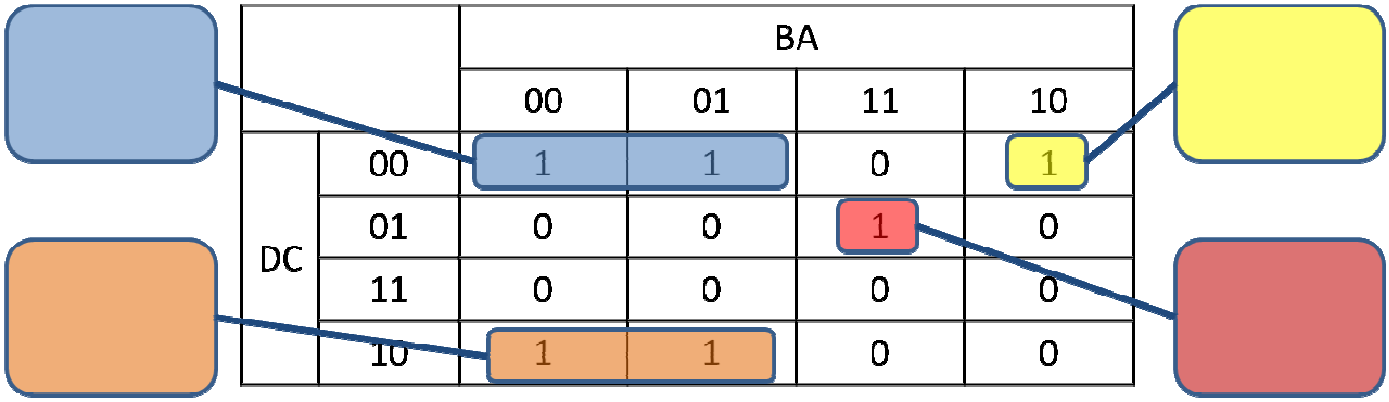
b. Determine the function number.

c. Write out the function in canonical sum of min terms form.

d. Without introducing new variables, produces an alternate but equivalent function.

2. Briefly explain what purpose(s) expressing a function in canonical sum of min terms form serves.

3. Consider the following Karnaugh map:



DC \ BA	00	01	11	10
00	1	1	0	1
01	0	0	1	0
11	0	0	0	0
10	1	1	0	0

a. Write the original function as a sum of minterms

b. Fill in the corresponding reduction for each indicated grouping.

c. Write the corresponding reduced function.

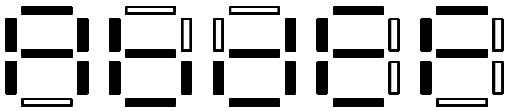
4. Consider the following function:

F = D'C'B'A' + D'CB'A' + D'C'BA + D'CBA + DCBA + DC'BA

a. Use a Karnaugh map to reduce the function. Clearly and unabigously indicate your reduction groupings.

b. Write out the reduced function

5. Suppose we want to modify a seven segment LED display to include hex values as follows:



Write out the decode function for segment 0:

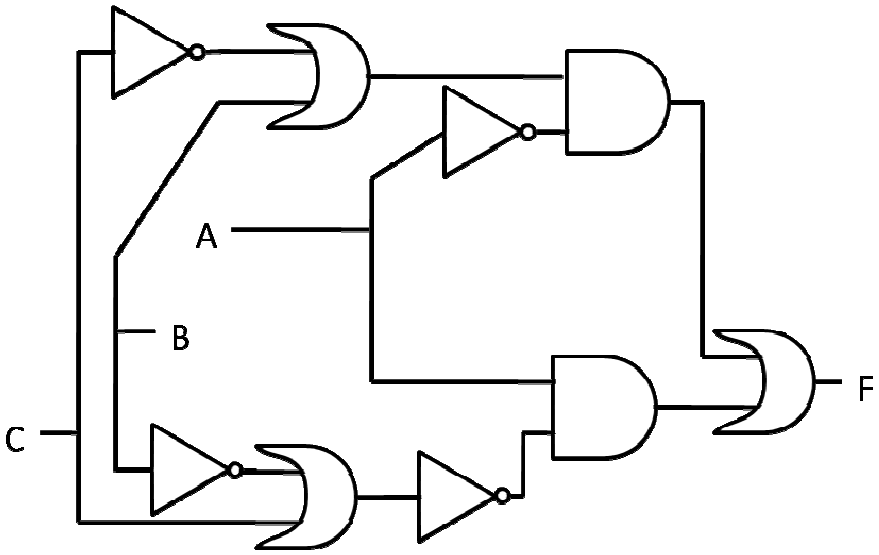
6. Implement equivalents for the following function using only NAND gates:

a. XOR

b. A ≤ B

7. Implement a 2bit by 2 bit multiplier (you may use AND, OR, NOT & NAND gates)

8. Write out the formula represented by the following circuit:



9. What information does a stack frame contain? Why do we use it to pass arguments to procedures?

10. Compare & contrast C calling convention with STDCALL convention.

11. Write some assembly code for a procedure which sums the contents of an array of WORDs as indicated below. Note: only output registers should be modified by the procedure.

Inputs:		Outputs:	
Offset to Array	EDX	Sum	EAX
Array length	ECX	Flags	Flags

12. Consider the following code listing, the right most column indicates the line's location in memory. Assume the stack starts out empty & diagram the stack (including EIP) where indicated. Note you may have more rows than you really need.

```
            Proc1 PROC
00401030    push        eax
            ; A) Diagram the stack here
00401031    call        Proc2 (401036h)
            Proc1 ENDP

            Proc2 PROC
00401036    push        ebx
            ; B) Diagram the stack here
00401037    call        Proc3 (40103Ch)
            Proc2 ENDP

            Proc3 PROC
0040103C    pop         ebx
0040103D    push        ecx
            ; C) Diagram the stack here
0040103E    push        0
00401040    call        ExitProcess
(401072h)
            Proc3 ENDP

main PROC
00401045    mov         eax,0Ah
0040104A    mov         ebx,0Bh
0040104F    mov         ecx,0Ch
00401054    mov         edx,0Dh
00401059    call        Proc1 (401030h)
0040105E    push        0
00401060    call        ExitProcess
(401072h)
```

A)

B)

C)

13. List the individual actions performed by the CALL instruction.