Gebze Technical University
Department of Computer Engineering
CSE 101 – Introduction to Computer Engineering
HW #1

1.Convert the following hexadecimal numbers to binary and binary numbers to hexadecimal. Separate each 4 bit from each other with a space character when converting to binary.

a)Hexadecimal---→ 8 A 9
Binary----------→ 1000 1010 1001

b)Hexadecimal---→ E F 3
Binary----------→ 1110 1111 0011

c)Binary----------→ 0001 1110 0001
Hexadecimal---→ 1 E 1

d)Binary----------→ 1111 1110 1101 1011
Hexadecimal--→ F E D B

2.Below is a mesage first coded in ASCII and then converted to hexadecimal. Decode the message and show our steps.

436F6D7075746572

43------→ C
6F------→ o
6D------→ m
70------→ p
75------→ u
74------→ t
65------→ e
72------→ r

3.Perform the mathematical operations below by converting each decimal into a 5-bit two's complement format. Check your results by doing the same operations in decimal format.Specify which of the operations causes an overflow.

a) 5 – 1

5------→ 00101

(-1)---→ + 11111

00100-------→ 4 (overflow doesn't exist.)

b) 5 – 11

5--------→ 00101

(-11)--→ + 10101

11010-------→ (-6) (overflow doesn't exist.)

4.Perform the following operations.

a ) 01001011 AND 10101011

01001011

AND 10101011

00001011

b) 01001011 OR 10101011

01001011

OR 10101011

11101011

c) 01001011 XOR 10101011

01001011

XOR 10101011

11100000

5.The followings are the instructions according to the machine language given in the appendix of your text book (Appendix C). Find the corresponding assembly commands.

a) 7123

Execute the OR logic operation between the bit pattern in register 2 and 3 and place the result in register 1.

b) 2BCD

Load the bit patternn CD in the register B.

6. Write an assembly program which obtains a 8 bit value by combining the first and last 4 bits of the memory cellss addressed with A0 and A1, respectively and writes this 8 bit
value into the memory address A2.

```
load R1 ,240  ;this step load F0 in register R1
load R2 ,15    ;this step load 0F in register R2
load R3 ,[$A0]
load R4 ,[$A1]
AND R2 ,R2,R4
AND R1 ,R1,R3
OR R5,R1,R2
store R5 , [0xA2]
HALT
```