

1. Contents of memory are:

#200: #ff      #201: #47      #202: #27      #203: #64  
#204: #61      #205: #79      #206: #21      #207: #00

a. The string beginning at #201 interpreted as ASCII characters is:

**G      '      d      a      y      !      null char**

b. The contents of the tetrabyte with address of #200, interpreted as an unsigned integer is:

$\#ff472764 = 4 + 16(6 + 16(7 + 16(2 + 16(7 + 16(4 + 16(15 + 16(15))))))$   
**= 4282853220.**

c. The contents of the tetrabyte with address of #200, interpreted as a signed integer is:

$\#ff472764 = 1111\ 1111\ 0100\ 0111\ 0010\ 0111\ 0110\ 0100$   
 $= -(0000\ 0000\ 1011\ 1000\ 1101\ 1000\ 1001\ 1100)$   
 $= -(12 + 16(9 + 16(8 + 16(13 + 16(8 + 16(11))))))$   
**= -12114076.**

2.

a. Suppose the contents of \$1 is #abcd9999ff876543 and \$2 is #eeee123443210101. Then the contents of \$0 after NXOR \$0,\$1,\$2 is

#abcd9999ff876543  
NXOR #eeee123443210101

= Not #45238badbca66442  
**= #badc745243599bbd.**

b. If we wanted to clear the leading bit of \$0, the MMIX instructions would be:

SETH	\$1,32768	//set contents \$1 as #8000...000
NXOR	\$1,\$1,0	//makes contents \$1 #7ffff..fff
AND	\$0,\$0,\$1	//keeps contents of \$0, but makes leading bit 0.

3.

a. The machine code of      STO \$0,\$20      is:  
**#ad001400.**

b. The MMIX instruction corresponding to machine instruction #23ff0010 is:  
**ADDU      \$255,\$0,16.**

4. Given the setup outlined in the instructions, the MMIX instructions that will replace the contents of octabyte C by the sum of a and b, and replace the contents of octabyte D by the XOR of a and b are:

LDO	\$0,A	//load contents of A (a) in \$0
LDO	\$1,B	//load contents of B (b) in \$1
SUM	\$2,\$0,\$1	//contents of \$2 are a+b
STO	\$2,C	//store a+b into address C
XOR	\$1,\$0,\$1	//contents of \$1 are a XOR b
STO	\$1,D.	//store a XOR b into address D.