**CSCI 360 Multiplication and Division**

**Multiplication**

* Multiply two 32-bit numbers to obtain a 64-bit result that is spread between an even-odd pair of registers
* **RX Format:** label M R,D(X,B)  
  + **R** is an even numbered register representing an even-odd pair of registers. The number to multiply must be in the odd numbered register
  + **D(X,B)** is the address of a fullword number to multiply by
* **RR Format:** label MR R1,R2
  + **R1** is an even numbered register representing an even-odd pair of registers. The number to multiply must be in the odd numbered register
  + **R2** is the number of a register containing the number to multiply by

**Example 1:** M 4,TWO where TWO DC F'2'

BEFORE: R4 = 05002037 R5 = FFFFFFFD (-3 in decimal)

AFTER: R4 = FFFFFFFF R5 = FFFFFFFA

This is the 2-register, or 64-bit, representation of -6

**Example 2:** MR 2,1

BEFORE: R1 = 00000004 R2 = 0000FFFF R3 = 00000005

AFTER: R1 = 00000004 R2 = 00000000 R3 = 00000014

This is the 2-register, or 64-bit, representation of 20

**Example 3:** MR 2,3 => squares the value in register 3

**Division**

* Divide a 64-bit number that is stored between an even-odd pair of registers by a 32-bit number.
* Results in a 32-bit quotient stored in the odd register and a 32-bit remainder stored in the even register.
* The remainder will ALWAYS have the same sign as the number being divided.
* Could cause a S0C9 - Fixed Point Divide Exception, which is usually caused by division by 0
* **RX Format:** label D R,D(X,B)  
  + **R** is an even numbered register representing an even-odd pair of registers. The dividend (ie. the number to be divided) must be spread between the two registers
  + **D(X,B)** is the address of a fullword number to divide by (ie. the divisor)
* **RR Format:** label DR R1,R2
  + **R1** is an even numbered register representing an even-odd pair of registers. The dividend (ie. the number to be divided) must be spread between the two registers
  + **R2** is the number of a register containing the number to divide by (ie. the divisor)

**EXAMPLE 1:** DR 2,8

Before: R2 = 00000000 R3 = 00000007 R8 = FFFFFFFE (-2 in   
 decimal)

After: R2 = 00000001 R3 = FFFFFFFD R8 = FFFFFFFE

REMAINDER QUOTIENT (-3 in decimal)

**EXAMPLE 2:** D 4,=F'2'

Before: R4 = 00000A00 R5 = 00000007

After: Error (SOC 9) because the quotient will not be 32 bits

**To get the desired results:**

M 4,=F'1' to zero out the even register  
D 4,=F'2'

After: R4 = 00000001 R5 = 00000003