Complete the following table (2) below for each instruction based on the memory contents in table 1.

Assume for each instruction that:

Register A = $01
Register B = $02
Register D = $0102
Register X = $1000
Register Y = $2000

Table 1

Memory address	Memory contents
$1000	$00
$1001	$01
$1002	$10
$1003	$04
$1004	$05
$1005	$04
$1102	$10
$1103	$02
….	…
$2000	$FF
$2001	$AA
$2002	$CC

Table 2

Instruction	Addressing Mode	Affected Reg	Contents of Reg	Effective Address
LDAA #00				
LDAB #FF				
LDAA $1000				
LDD $1000				
LDX $2000				
LDX $2001				
LDD 0,Y				
LDAA 0,X				
LDAB 4,X				
LDD A,Y				
LDAA B,X				
LDAA [D,X]				

Problem 2

Write a subroutine that moves 16 bytes of data from a source range in the memory map to a destination range. Assume that X has the first address of source range and that X + 15 is the final address in the source range. Y – similarly for the destination - has the initial destination address and Y + 15 is the final destination address.

Problem 3

Write a subroutine that takes an eight bit unsigned number passed in register A, squares the number and returns the result in register A. The subroutine must check that the input value is small enough that the square is less than 256 and if not A should be cleared as an indication of overflow.

Problem 4

Write a short program that will light up one of the eight LEDs on Port B (assume Dragon 12 board) and have the light race from right to left and then left to right – seemingly bouncing back and forth. The program should run forever .