PROGRAMS ON ARITHMETIC AND LOGICAL INSTRUCTIONS

Write an assembly program to multiply two 32 bit numbers

Objectives

- Identify and use the instructions required to perform multiplication, division and logical operations
- Debug and trace the programs

```
        AREA    RESET, DATA, READONLY
        EXPORT  __Vectors

  __Vectors
        DCD  0x40001000     ; stack pointer value when stack is empty
       DCD  Reset_Handler  ; reset vector
       ALIGN
       AREA mycode, CODE, READONLY
       ENTRY
       EXPORT Reset_Handler
  Reset_Handler
        LDR R1, =VALUE1
        LDR R2,=VALUE2
        UMULL R3, R4, R2, R1
        LDR R2, =RESULT
        STR R4, [R2]

        ADD R2, #4
        STR R3, [R2]  ; store result in memory
STOP
        B STOP
        VALUE1 DCD 0X54000000 ; First 32 bit number
        VALUE2 DCD 0X10000002 ; Second 32 bit number
        AREA data, DATA, READWRITE
        RESULT DCD 0
```

```
;pointer to the first value1
;pointer to the second value
;Multiply the values from R1 and R2 and store
 ;least significant  32 bit number into R3 and most
 ;significant  32 bit number into R4.
```

Note: If the result within 32 bits, use MUL instruction.

Lab Exercises:

1. Write a program to multiply two 32 bit numbers using repetitive addition

   Hint: If two numbers are in R0 and R1 Registers then use following algorithm

   ```
   Sum=0;
   do { sum=sum+R0; R1--;      ;Use ADDS instruction for addition and use  ADD
                               ;instruction to increment a register by 1
   if carry then
   R2++;                       ;Increment carry value by one.
    } while(R1!=0);            ;Use Compare instruction to check greater
                               ;than or not. And Brach instructions for loop
   Result= R2 and R0
   ```

2. Repeat the above program for BCD multiplication
3. Find the sum of ‘n’ natural numbers using MLA instruction.
4. Write an assembly language program to find GCD of two numbers

   Hint:

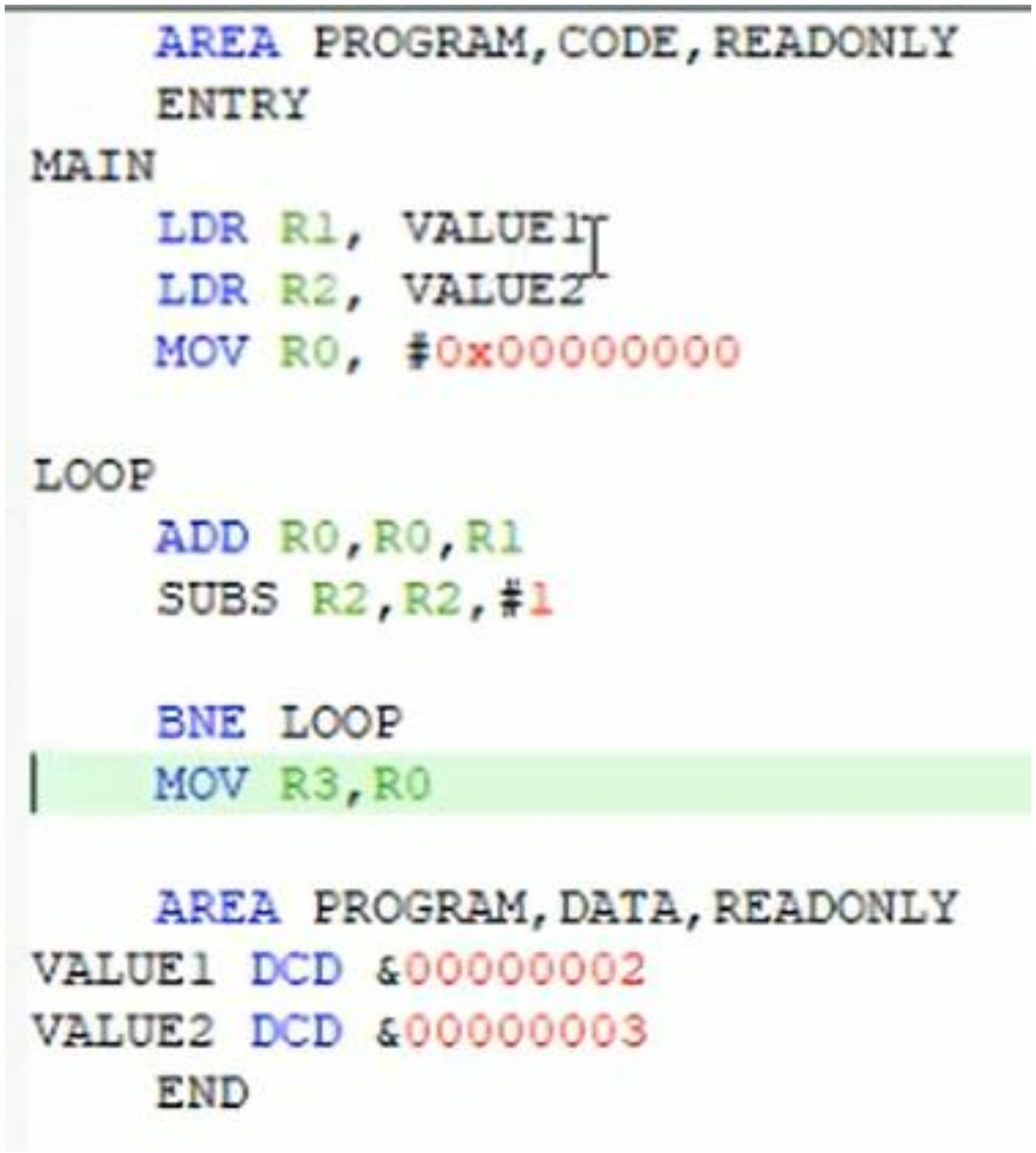
   ```
    While(a!=b)
   {
           If(a>b)
           a=a-b;
           else
           b=b-a;

   } Return (a);
   ```

5. Write an assembly language program to find LCM of two numbers

Additional Exercises:

1. Write an assembly language program to generate Fibonacci series.
2. Write an assembly language program to divide a 32-bit number by 16-bit number by repetitive subtraction
3. Check whether a given number is even or odd.



```
    AREA PROGRAM,CODE,READONLY
    ENTRY
MAIN
    LDR R1, VALUE1
    LDR R2, VALUE2
    MOV R0, #0x00000000

LOOP
    ADD R0,R0,R1
    SUBS R2,R2,#1

    BNE LOOP
    MOV R3,R0

    AREA PROGRAM,DATA,READONLY
VALUE1 DCD &00000002
VALUE2 DCD &00000003
    END
```