```
        AREA IntegerAddition, CODE, READONLY                                // Name this block of code.

        EXPORT          IntegerAddition

/* Integer Addition Subroutine */

/*
   This subroutine adds 1,000 integers starting at the memory
   address ARRAY and stores the result in the memory address
 */

IntegerAddition
        STMFD   SP!,{R0-R6,LR}                                  // Store registers.

        LDR R0, =0                                              // offset = 0
        LDR R1, =0                                              // count = 0
        LDR R2, =0                                              // sum = 0

        LDR R3, =ARRAY                                          // integersAddress
        LDR R4, =SUM                                            // sumAddress (2 x 32 bit)

while
        CMP R1, #1000                                           // while(count < 1000)
        BGE endWhile

        ADD R1, R1, #1                                          // count++
        LDR R5, [R3, R0]                                        // val = loadInteger()
        ADDS R2, R2, R5                                         // sum += val
        BCC noCarry                                             // if(carryOccurred)

        LDR R5, [R4]                                            // load significant part of SUM
        ADD R5, R5, #1                                          // sigPart ++
        STR R5, [R4]                                            // storeUpdatedVal()

noCarry

        STR R2, [R4, #4]                                        // update less significant part
        ADD R0, R0, #4                                          // offset ++
        B while

endWhile
        LDMFD           SP!, {R0-R6,PC}^                        // Restore registers and return.

        END
```