```
    1           AREA    Program, CODE, READWRITE
    2           ENTRY
    3   ;       R0 - Array Current Number
    4   ;       R1 - Index
    5   ;       R2 - Maximum
    6   ;       R3 - Minumum
    7   ;       R4 - Sum
    8   ;       R5 - Length
    9   ;       R6 - Average
   10   ;       R7 - Counter
   11           LDR     R1, =Array      ;       Load R1 with starting address of array
   12           LDR     R7, Len         ;       Load R7 with length of array
   13           MOV     R5, R7          ;       Copy length of array to R5 (used for average)
   14           LDR     R2, [R1]        ;       Initialize Minimum to first element
   15           LDR     R3, [R1]        ;       Initialize Maximum to first element
   16           LDR     R4, [R1]        ;       Initialize Sum to first element
   17   loop1   SUBS    R7, R7, #0x1    ;       Decrement counter
   18           BEQ     end1            ;       End loop if counter = 0
   19           ADD     R1,R1,#+4       ;       Incrememnt array pointer
   20           LDR     R0, [R1]        ;       Load next number to R0
   21           CMP     R0, R2          ;       Compare current with maximum
   22           MOVGT   R2, R0          ;       Replace maximum if current number is greater
   23           CMP     R0, R3          ;       Compare current with minimum
   24           MOVLT   R3, R0          ;       Replace minimum if current number is lesser
   25           ADD     R4, R4, R0      ;       Add current number to sum
   26           B       loop1           ;       Go back to loop
   27   end1    EOR     R6, R6, R6      ;       Clear R6 which will store average
   28   loop2   CMP     R4, R5          ;       Compare divisor with dividend
   29           BLT     end2            ;       End loop if divident less than divisor
   30           SUB     R4, R4, R5      ;       Subtract divisor from dividend if divisor is less
   31           ADD     R6, R6, #+1     ;       Add 1 to quotient of divisor is less
   32           B       loop2           ;       Loop if divisor is less
   33   end2    STR     R2, Max         ;
   34           STR     R3, Min         ;
   35           STR     R6, Avg         ;       Store the results to memory
   36   end3    B       end3
   37
   38           AREA    Table, DATA, READWRITE
   39   Array   DCD     10,30,20
   40   Len     DCD     3
   41   Min     DCD     0
   42   Max     DCD     0
   43   Avg     DCD     0
   44           END
```