

Lab1: Instruction to ARM Programming

Group66

Gengyi Sun 260768270

Liang Zhao 260781081

1. PRAT1: Finding a range of a set of data

```
_start:
    LDR R7, =RANGE
    LDR R2, [R7,#4] //Hold number of elements in list
    ADD R3, R7, #8//Points to the location of first number Loop counter
    LDR R0, [R3]
    MOV R1, #0

LOOP:    SUBS R2, R2, #1
        BEQ DONE
        LDR R4, [R3]
        ADD R3, R3, #4

CMPMIN:  CMP R0, R4
        BGT ISMIN
        B CMPMAX

ISMIN:   MOV R0, R4
        B CMPMAX

CMPMAX:  CMP R1, R4
        BLT ISMAX
        B NOTHING

ISMAX:   MOV R1, R4
        B NOTHING

NOTHING: SUB R6, R1,R0
        B LOOP

DONE:    STR R6, [R7]

END:     B END
```

Figure 1

Instead of using codes from part1.s, which can only get the maximum or the minimum, this algorithm finds the maximum and minimum in one loop.

R0 holds for the minimum and R1 holds for the maximum. Before go in the loop, load the first number to R0, and load number 0 to R1. Since there must exists some number that is smaller than the first number in the list or larger than 0.

In the loop, compare the loaded number with R0 and R1 respectively. Replace R0 if the number is smaller than R0, and replace R1 if the number is greater than R1.

Finally when the list is exhausted, the range of the list is obtained by subtracting R1 by R0. Store its value in memory location.

● PART2: Maximum and minimum values of an algebraic expression

`_start:`

```
LDR R9, =MAX
LDR R10, =MIN

LDR R0, [R10, #4]
LDR R1, [R10, #8]
LDR R2, [R10, #12]
LDR R3, [R10, #16]
```

Figure 2: Load all numbers

```
ADD R4, R0, R1
ADD R5, R2, R3
MUL R6, R4, R5
MOV R4, #0
MOV R5, #0

ADD R4, R0, R2
ADD R5, R1, R3
MUL R7, R4, R5
MOV R4, #0
MOV R5, #0

ADD R4, R0, R3
ADD R5, R1, R2
MUL R8, R4, R5
MOV R4, #0
MOV R5, #0
```

Figure 3: Compute All Combinations

We implement this question by listing all the possible combinations of $(a + b) * (c + d)$, which are:

- | | | |
|----------|----------|----------|
| 1. ab cd | 2. ba dc | 3. ac bd |
| 4. ca db | 5. ad bc | 6. bd cb |

We simplify these by merge these equivalent combinations to: (figure 3)

- | | | |
|----------|----------|----------|
| 1. ab cd | 2. ad bc | 3. ac bd |
|----------|----------|----------|

Therefore, there are three combinations we need to consider, we calculate the Maximum and Minimum values of this expression by comparing the products of these three combinations(figure 4).

```
CMP R6,R7
MOVGT R5, R6
MOVGT R4, R7
MOVL T R5, R7
MOVL T R4, R6
CMP R5, R8
MOVL T R5, R8
CMP R4, R8
MOVGT R4, R8
STR R4, [R10]
STR R5, [R9]
```

Figure 4: Compare for Max&Min

Possible improvements: this is a hard coding and can only handle the situation of $(a + b) * (c + d)$, a better way of doing this is to create a method which can find all possible combinations of $X(S - X)$, calculate their products and find the max and min, but it is much more complicated and unnecessary for this given requirement of this lab.