CS1021 Tutorial #6
Using Memory

1 String Length

Assume that a NULL-terminated string of ASCII characters is stored in memory beginning at the address contained in R1. Design and write an ARM Assembly Language program to compute the length (number of characters) in the string, excluding the NULL character. Store the result in R0.

2 String Duplication

Design and write an ARM Assembly Language program to create a new string in memory that is a duplicate of an existing string, also stored in memory.

Assume that R1 contains the address of the existing string and R0 contains the address where the new string should be stored, as illustrated in the template code on the following page.

```
1  start
2      LDR R1, =stringA     ; Load start address of stringA
3      LDR R0, =stringB     ; Load start address of stringB
4
5      ;
6      ; your program to duplicate the string
7      ;
8
9  stop
10     B   stop
11
12     AREA     TestData, DATA, READWRITE
13
14     ; Test Data
15
16 stringA
17     DCB "motor",0    ; NULL terminated test string
18
19 stringB
20     SPACE   128 ; 128 bytes of reserved memory to store the duplicated string
21
22     END
```

3 Pseudo-code to ARM Assembly Language

Translate the following pseudo-code extract into ARM Assembly Language. Assume that `a`, `b`, `c`, `N` and `address` are unsigned values stored in `R0`, `R1`, `R2`, `R3` and `R4` respectively. The syntax `Memory.word[address]` is intended to represent loading the word-size value from memory at the specified address.

```
a=0;
c=0;
while (a < N) {
    address = b + (a * 4);
    c = c + Memory.word[address];
    a = a + 1;
}
```

What does the program above do?

4 String Reversal

Design and write an ARM Assembly Language program to create a new string in memory that is the reverse of an existing string, also stored in memory.

For example, if the original string was "hello", your program should store the string "olleh" in memory.

Assume that `R1` contains the address of the existing string and `R0` contains the address where the new string should be stored.

5 Palindromes

A *palindrome* is a word or sentence that reads the same in both forwards and backwards direction. For example "kayak" and "racecar" are well-known examples of palindromes.

Design and write an ARM Assembly Language program that will determine if a NULL-terminated string stored in memory is a palindrome. Assume that the start address of the string is stored in R1. Your program should store 1 in R0 if the string is a palindrome and 0 if it is not.