

CS1021 Lab 3

Greatest Common Divisor

- Q1 The greatest common divisor (GCD) of two positive integers a and b is the largest integer that divides both a and b without a remainder. For example, the GCD of 24 and 32 is 8.

Convert the following pseudo code for calculating the GCD of a and b into ARM Assembly Language.

```
while (a != b) {
    if (a > b) {
        a = a - b;
    } else {
        b = b - a;
    }
}
```

Assume that a and b are unsigned integers stored in R0 and R1 respectively. The result will be stored in a (and b). Test your program by calculating $\text{gcd}(24, 32)$, $\text{gcd}(2415, 3289)$, $\text{gcd}(4278, 8602)$ and $\text{gcd}(406, 555)$.

64-bit Arithmetic

- Q2 Using **unsigned 64-bit** integers, write an ARM Assembly Language program to compute the n^{th} Fibonacci number. The n^{th} Fibonacci number is defined recursively as:

$$F_n = F_{n-1} + F_{n-2} \text{ where } F_0 = 0 \text{ and } F_1 = 1.$$

Use R6 for n and compute F_n in R0:R1. Test your program by computing F_{48} and F_{64} .

- Q3 Using your answer to Q2 as a starting point, write an ARM Assembly Language program to calculate the largest possible Fibonacci number using **64-bit signed** arithmetic. Make sure that you report the values of n and F_n (in hexadecimal and decimal) in your project submission.

Hint: If the maximum signed 64-bit integer is MAX, it is possible to terminate the loop generating the Fibonacci numbers using the following condition:

```
if (MAX - Fn-1 < Fn-2)          // Fn-1 + Fn-2 will be greater than MAX...
    break;                       // therefore Fn-1 is the largest Fibonacci number
```

This lab will count towards your final CS1021 coursework mark. Submit your solution lab3.s and evidence that your program works (eg. use the Windows Snipping tool to get screen shots of the Keil IDE which show that your program generates the correct results) via Blackboard no later than **9am on Fri 19-Oct-2018**.