**Purpose:** To see how to implement libc functions in ARM and learn how these functions affect the registers.

**Overview:** You will create a program that finds the sine, cosine, and tangent of a given number. You will then print out the answer onto the console. This program will require the use of floating-point registers and VFP instructions to work properly. The implementation for the sine function is given and should be used to help with the implementation of the cosine and tangent functions.

**Directions:**

1. Follow the example given with the sinf function. You will follow the method used in this example to implement cosf and tanf. Note that printf must be implemented differently
2. Load r0 with addr\_value1
3. Load the value at addr\_value1 into the VFP single-precision register of s0
4. Branch with link to the cosf function
5. The result of the cosf function will be placed in s0. Convert this value into 64-bit floating-point number and store that value in d5
6. Move the 64-bit value in d5 into the registers r4 and r5
7. Do the same for the tanf function
8. To use printf, we must first pop the values of r2 and r3 we pushed earlier back into r2 and r3
9. Push registers r4 to r7 to follow AAPCS standards
10. Load r0 with addr\_sinout
11. Branch with link to printf
12. Pop registers r4 to r7
13. If you did everything correctly, the sine should be **0.565956**, the cosine should be **-0.824435,** and the tangent should be **-0.686477**.