Programming Lab Policies:

- Labs that fail to compile, or do not terminate correctly, will receive a zero.
- Labs that fail to compile, or do not terminate correctly, may not be resubmitted for a grade. This includes instances where students did not upload the correct file for grading.
- **Students must make a credible attempt to pass all programming labs to receive a passing grade in the course.**

Write ARM assembly implementations for all of the functions below. The functions must be present in a single .s file. Your function/procedure names must be identical to that presented below, as your implementations will be tested with generic C code used by the TAs.

All of your functions must return a value such that the program will run to completion with no segmentation faults. If a function cannot be successfully implemented, it still must return a valid value: **no function may be omitted**. Attempting to omit a function will result in a compile error.

Submit your assignment via the submission link on Canvas. The name of this file should be **lab#_lastname_loginID.s** where # is the number of the lab assignment. Example: If your name is John Doe and your login ID is jxd1234, your submission file name must be "lab#_Doe_jxd1234.s".

All questions worth ten points.

1. void stringCopy(char* strTo, char* strFrom)
// copies strFrom to strTo

2. void stringCat(char* **strFrom**, char* **strTo**)
// adds **strFrom** to end of **strTo**

3. int32_t sumS32(int32_t x[], int32_t count);
// returns sum of the values in the array (x) containing count entries

4. int32_t sumS16(int16_t x[], int32_t count);
// returns sum of the values in the array (x) containing count entries

5. uint64_t sumU32_64(uint32_t x[], uint32_t count);
// returns the 64-bit sum of the values in the array (x) containing count entries

6. uint32_t countNegative (int16_t x[], uint32_t count);
// returns number of negative values in the array (x) containing count entries

7. uint32_t countNonNegative (int16_t x[], uint32_t count);
// returns number of non-negative values in the array (x) containing count entries

8. uint32_t countMatches(char str[], char toMatch);
// input: char (toMatch) containing the character to match in the string (str)
// output: returns the number of occurrences of toMatch in str, or 0 if not found

9. int32_t returnMax(int16_t x[], uint32_t count);
// returns the maximum value from the integer array (x) with count entries

10. int32_t returnMin(int16_t x[], uint32_t count);
// return the minimum value from the integer array (x) with count entries

Tip: Remember the concept of sign extension. When working with 16-bit integers, there are both signed and unsigned load commands.