CS 382: Homework Assignment 2
Due: October 8, 11:55pm

Collaboration Policy. Homeworks will be done **individually**: each student must submit their own answers. It is acceptable for students to collaborate in understanding the material but not in solving the problems or programming. Use of the Internet is allowed, but should not include searching for existing solutions.

Under absolutely no circumstances code can be exchanged between students. If some code was shown in class, it can be used, but it must be obtained from Canvas, the instructor or the TA.

Assignments from previous offerings of the course must not be re-used. Violations will be penalized.

Late Policy. No late submissions will be allowed without consent from the instructor. If urgent or unusual circumstances prohibit you from submitting a homework assignment in time, please e-mail me.

Deliverable. A single **pdf** file on Canvas.

Problem 1 (15 points) Consider the following high level code. Assume the 12-element array A is initialized before it is used, and that register X19 holds the base address of A. Convert the code to ARM. All numbers used are 8-byte integers.

```
for (i = 0; i < 12; i ++)
    A[i] = A[i] - i;
```

Problem 2 (15 points) For the following C statement, write the corresponding ARM assembly code. Assume that the variables i, and j are assigned to registers X20 and X21, respectively. Assume that the base addresses of the arrays A and B are in registers X22 and X23, respectively. All numbers used are 8-byte integers.

```
B[6] = A[i + j];
```

Problem 3 (20 points) Convert the following code into ARM assembly. Assume a and b are signed numbers and have been placed in registers X19, and X20 respectively:

```
if (a < b) a = a + b;
else a = 8 * b;
```

Hint: again, you don't need to use multiplication instruction. Using it will not make you lose points, but it will make you look bad ;)

Problem 4 (15 points) Assume X0 holds the value 0010 0100. What is the value of X1 after the following instructions? **Explain** your answer.

```
        CMP   X0, #20
        B.GE  ELSE
        B     DONE
ELSE:   LSR   X1, X0, #2
DONE:
```

Hint: `CMP X, Y` is similar to `SUBS Z, X, Y` — `CMP` subtracts `Y` from `X`, and sets the condition codes, but doesn't store the result to any register.

Problem 5 (15 points) Consider the following ARM loop:

```
        ADD  X10, XZR, #1
        ADD  X11, X10, #0
LOOP:   SUBS X12, X11, #5
        B.GE DONE
        ADD  X11, X11, #1
        LSL  X10, X10, #1
        B    LOOP
DONE:
```

What is the final value in register X10? **Explain** your answer.

Problem 6 (20 points) Consider the following high level code and convert it to ARM assembly. All variables declared in a function are local. Assume that execution will start from the first line of your code and will end when `caller()` returns. All numbers used are 8-byte integers.

```
long int caller(void) {
    long int x = 2;
    long int y = 3;
    long int z;
    z = addition(x, y);
    return z;
}

long int addition(long int a, long int b) {
    return a + b;
}
```