

Cmpe344 Fall 2021

Experiment #1: Arithmetic operations on arrays

In this experiment, you will write a MIPS program that has two arrays with **byte** elements in its data segment. The program should subtract the second array from the first, element-wise, and store the results into the first array, overwriting it. The second array should remain the same. A byte with the value 0 will mark the end of both arrays.

You can assume that the input arrays' lengths will be equal, and that the input values and the calculated differences will be representable as signed bytes; we will ensure these in our test cases.

For example, if the input arrays are;

41, 82, 25, 19, -7, -1, 6, 2, 0
3, 2, 1, -9, -6, -3, 14, 15, 0

when the program finishes, they should become;

38, 80, 24, 28, -1, 2, -8, -13, 0
3, 2, 1, -9, -6, -3, 14, 15, 0

Questions

Note that the input values or their sums might be too large or too small to store as **signed bytes** for some inputs.

In the following table, a_i and b_i denote the values of the i^{th} elements of the two arrays. Fill in the table for the given a_i :

1. Indicate whether it is possible to store a_i as a signed byte.
2. Give the minimum and maximum values for b_i such that $a_i - b_i$ can still be stored back into a_i as a signed byte.

a_i	Possible? [yes/no]	Minimum b_i value	Maximum b_i value
-180	No	-	-
-50	Yes	-128	78
50	Yes	-77	127
180	No	-	-