

ASCII to Integer Conversion

Requirements

1. Mac, Windows or Linux PC
2. Java standard edition (SE) version 9 or higher.
3. MARS Java program

Introduction

In this lab, you will use *MARS* (**M**IPS **A**ssembler and **R**untime **S**imulator) to write and run a program that reads in a string of ASCII characters and converts them to Integer numbers stored in an array.

If you have any problems running MARS, refer to the instructions in Lab 1.

There are two different ways to convert the number into ASCII, subtraction and “masking”.

- If your student ID ends in an odd number, then use subtraction.
- If your student ID ends in an even number, then use masking.

There is a complete integer to ASCII reference table in the textbook.

Exercise 1

Write a program that:

1. Inputs a 1x8 vector of single-digit integers
2. Stores them into an 8-entry 32-bit Integer array, “V”.

It is not completely trivial to do this given the Syscalls available and the desired input format.

Hint: Use *Read String* and not *Read Integer*, then convert from ASCII to integer before storing into the integer array, “V”. Use the ASCII table in the book to determine how to convert from ASCII to integer (there are two ways, both very easy, select the method as per the introduction).

After storing the integers in the array:

1. Read the same values using *Read Integer* and store them in a 32-bit integer array, “VPrime”.
2. Subtract the two arrays integer by integer and put the results into a third 32-bit integer array, “VCheck”.
3. Sum all the values in VCheck and using *Write Integer*, display the result.

When you run the program, the input should look something like this with a space between numbers:

Input V: 1 4 0 2 7 3 8 4

(this is just an example vector; it can be any string of single digit integers)

Input VPrime:

1
4
0
2
7
3
8
4

(Where the integers: 0 1 2 3 4 ... 8 9 or whatever vector values the user wants to input are input by the user on the “console.”)

And the output will look like:

Check Result: 0

(Upload the program to Canvas in runnable form.)