CSC252 ASM1 Report
Kory Smith

- ˆ First item for your report - The address of some variable. Now, look more closely at what LA did. It put the address of the variable into the register you chose. Give the address in hexadecimal. If all of the registers are in decimal instead of hexadecimal, look at the Settings menu to change them to hexadecimal.

0x10010094

- Second item for your report - The two instructions that make up LA Next, look at the two instructions which make up the LA pseudoinstruction (the first is LUI). Write down, for your report, exactly what MARS shows in the Basic column of the Text Segment for these two instructions. The register names won't look familiar - this column uses register numbers instead of names - but the second instruction should have a number that you've seen before.

Lui $1,0x00001001
Ori $16,$1,0x00000094

- Third item for your report - Hex encodings Finally, write down the actual 32-bit hex encodings of these two instructions (which you can find in the Code column). Do you see the constants used by these instructions stored inside those encodings somewhere?

32-bit hex encoding for Lui $1,0x00001001 = 0x3c011001
32-bit hex encoding for Ori $16,$1,0x00000094 = 0x34300094
I observed that the actual 32-bit hex encoding for all LA instructions is 0x3c011001

2.1 Strings and data:

- Entire String constant = RED: .asciiz "red: "
- ASCII codes for the First 4 characters of orange in hex = 6F 72 61 6E
- Address of string 'orange' in memory = 0x10010054
- The word in memory which contained the expected ASCII codes. = 0x6e61726f

Reflection:

I learned a lot from this exercise along the lines of how objects are stored in memory and processed by computer. I enjoyed learning the MIPS assembly language and I hope to use it at some point in my future. Even if I never write assembly code again this exercise gave me a better understanding of registers, addresses, and objects memory allocation. One question I had regarding this report was why hexadecimal encodings of string addresses are stored backward.