**Homework 1119: MIPS Instructions**

Last modified: 12 November 2021

Due: 19 November, 5:00 pm

**General Instructions**

Answer the questions below in a text document, containing only plain ASCII characters, with lines limited to 78 characters in length. Clearly number each response. Make sure you put your name at the top of the document!

Comment each MIPS statement you write so that the person reading your code knows what you are trying to accomplish. Your comments can contain pseudo-C statements when it makes sense. In this homework, lowercase letters are variables stored in *registers*, and uppercase letters are array names for values stored in *RAM*. Variables a through d are always signed integer values, while i is always an unsigned integer value. Make the following assumptions about variable locations:

|  |  |
| --- | --- |
| **variable** | **register** |
| a | $s0 |
| b | $s1 |
| c | $s2 |
| d | $s3 |
| i | $s4 |
| A | $s6 |
| B | $s7 |

Style rules:

* all hex letters in lowercase
* space after a comma, no space before a comma
* space on both sides of binary operators

Problem 1. Translate the following C statements into MIPS assembly language. Assume that all variables are 32-bit signed integers. Translate the statements exactly as written; do not use any algebraic rules to simplify the expressions, and do not rearrange the order of operations unless MIPS requires you to do so. Use t registers for temporary values as needed.

a. a = b + c + a;  
b. b = (a + b) - (a - b);  
c. d = a - (a + b) + (a + b - c) - d;

Problem 2. Translate the following C statements into MIPS assembly language. Assume A and B are arrays of 32-bit signed integers.

a. A[1] = a;  
b. b = A[2];  
c. c = A[i];  
d. A[i] = B[i];  
e. A[i - 1] = B[i + 1];

Problem 3. Translate the following C statements into MIPS assembly language. Assume that all variables are 32-bit signed integers. Translate the statements exactly as written; do not use any algebraic rules to simplify the expressions, and do not rearrange the order of operations unless MIPS requires you to do so.

a. b = ++a;  
b. c = ++a - b++;  
c. c = (a + b + c) + (c + -1 - d) - (a - -5);

Problem 4. Translate the following MIPS instructions into machine code. Give your answers in hex.

a. addu $s0, $s2, $s4  
b. sub $t1, $zero, $t1  
c. lw $fp, 0($zero)  
d. sw $ra, -16($t7)

Problem 5. Disassemble the following machine code into MIPS assembly language statements.

a. 0x00028821  
b. 0x34020004  
c. 0x0211082a  
d. 0x27a50004

Problem 6. Convert the following C code to MIPS assembly language. Do not use any procedures for this; code the statements as written. Assume all variables are signed 32-bit integers.

a.

if (a != b)

{

a = b;

}

else

{

b = -a;

}

b.

while (a + 1 == 2 \* b)

{

a \*= 3;

}

c.

while (a <= b + 1)

{

b = 4 \* (a - 100);

a++;

}

By 5 pm on Friday, `9 November, submit your text file to the [homework submission](https://borax.truman.edu/250/submit.php) page.