



# Pre-laboratorio 1: Instrucciones MIPS y Simulación en MARS

## Objetivos de aprendizaje

- Predecir funcionamiento de un programa MIPS
- Traducir programas escritos en un lenguaje de alto nivel a MIPS
- Escribir programas MIPS que usan instrucciones aritméticas, de salto y memoria
- Usar MARS (un IDE para MIPS) para escribir, ensamblar y depurar programas MIPS

## Entrega

Sube los siguientes entregables a través del Google Classroom correspondiente a tu sección de laboratorio:

1) Antes de comenzar la sesión de laboratorio, sube las respuestas de la actividad de Pre-laboratorio como un archivo PDF.

## Actividades Pre-Laboratorio

### 1. Instalación de MARS

Instala en tu computador el simulador MARS de MIPS:

<http://courses.missouristate.edu/kenvollmar/mars/>

Nota que se requiere la plataforma JAVA, que se puede descargar aquí:

<https://www.oracle.com/java/technologies/javase-downloads.html>

### 2. Entendiendo MARS

Mira las dos primeras lecciones de los videos tutoriales de MARS (los subtítulos tienen traducción automática al español, no excelente, pero se entiende muy bien):

- <https://www.youtube.com/watch?v=oMI2H8PFukk>
- <https://www.youtube.com/watch?v=22xEtqRivxg>

No es necesario entender cada detalle de estos 2 videos. En cambio, enfócate en responder las siguientes preguntas acerca de MARS:

- ¿Cómo se *ensambla* un programa MIPS?
- ¿Cómo se *ejecuta* un programa en MIPS?
- ¿Cómo detendrías la ejecución en cierta línea que no es necesariamente la última?
- ¿Dónde encontrarías el valor actualmente almacenado en el registro \$s0?



- ¿Dónde encontrarías el valor actualmente almacenado en la dirección de memoria 0x10010024?

### Predecir el funcionamiento de programas MIPS

Para cada programa, responde la pregunta sobre el resultado. No utilices *software* en esta parte, sino que desarrolla “a mano” escribiendo el resultado en papel.

1. ¿Cuál es el valor en los registros \$t1 y \$t0 al terminar el programa?

```
addi $t0, $zero, -4  
add $t1, $t0, $t0
```

2. ¿Cuál es el valor en los registros \$t1 y \$t0 al terminar el programa cuando \$t2 almacena un 2? ¿Y cuando \$t2 almacena un 0?

```
addi $t0, $zero, 2  
add $t1 $t0 $t2  
beq $t0, $t1, A  
addi $t1, $zero, 1
```

3. ¿Cuál es el valor en los registros \$t2, \$t1 y \$t0 y las direcciones de memoria 0x10010000 y 0x10010004 al terminar el programa?

```
addiu $t0, $zero, 0x10010000  
addi $t1, $zero, 5  
sw $t1, 0($t0)  
lw $t2, 0($t0)  
addiu $t0, $t0, 4  
sw $t2, 0($t0)
```