

## Proyecto II (15%)

### Objetivos Generales:

Desarrollar un programa para imprimir por consola las instrucciones de un programa ensamblado y hacer un recuento de la cantidad de instrucciones de cada tipo.

### Objetivos Específicos:

- Adquirir destrezas en el lenguaje de bajo nivel del simulador SPIM
- Adquirir destrezas en el uso de estructuras de datos en el lenguaje de bajo nivel
- Aprender a programar subrutinas o funciones en el lenguaje de bajo nivel
- Aprender a usar las convenciones de SPIM para el uso de subrutinas/funciones

### Enunciado

En este proyecto, se desea que Ud. desarrolle un programa capaz de leer y mostrar por consola un programa cargado en memoria escrito con un conjunto reducido de instrucciones MIPS.

El conjunto de **instrucciones** consiste en las siguientes instrucciones MIPS:

- Lectura / Escritura: lw, sw (usando registro, etiqueta o etiqueta + registro)
- Operaciones aritméticas: add, addu, addi, sub, subu, and, or, xor, nor
- Bifurcaciones: beq, bne, blez, bgtz
- Saltos: j, jal

### Registros

El programa que leerán solo operará con los 32 registros de propósito general.

### Ambiente

Para el funcionamiento de su lector, el programa a traducir estará en la área de texto de un archivo exceptions.s modificado. La primera línea del programa estará indicada por la etiqueta \_\_principal, y la última instrucción del programa será un break 0 (que no debe mostrarse).

Ver ejemplo exceptions2010.s en aula virtual.

Adicionalmente, después de completada la lectura de un programa y antes de ejecutar el retorno (via "jr \$ra"), debe imprimir las siguientes estadísticas:

- El total de instrucciones tipo R leídas
- El total de instrucciones tipo I leídas

- El total de instrucciones tipo J leídas

Por ejemplo, si leen 100 instrucciones de las cuales 57 son tipo-R, 32 son tipo-I, y 11 son tipo-J., su simulador deberá imprimir lo siguiente:

Tipo-R: 57

Tipo-I: 32

Tipo-J: 11

Algunas instrucciones cuando son ensambladas se traducen en dos o más instrucciones de máquina. Estos casos se deben detectar y se debe mostrar sólo la instrucción original, tanto en su impresión por consola como en el conteo del número de operaciones de cada tipo.

Ejemplos

addi

0x20080001 -> addi \$t0, \$zero, 1

beq

0x11600005 -> beq \$t3, \$zero, [0x00400034]

break

0x0000000d -> (fin de programa)

## Código

El código debe tener una cantidad adecuada de comentarios; dentro de los comentarios debe estar bien clara la planificación de los registros, y el programa deberá estar correctamente sangrado.

Deben cumplir con las convenciones que se verán en clase para el manejo de subrutinas.

## Entrega

La entrega del proyecto es el día martes 8 de Noviembre antes de las 23:30h.

Ud. deberá colocar el archivo lector.s con su proyecto en Aula Virtual, para lo cual deberá crear el directorio Proy2 dentro de la carpeta documentos de su grupo. Note que debe estar inscrito en algún grupo en Aula Virtual para poder optar a esta opción. En este directorio colocará el archivo fuente correspondiente a su lector. No se corregirán proyectos que no sean colocados. Adicionalmente, el día 12 de noviembre, en horario de laboratorio se realizará la ejecución presencial con los grupos.

En el código se deberá indicar el número de grupo y los nombres de los integrantes del grupo.