



Nombre y Apellido ..... LEGAJO .....

**Ejercicio 1.** Desarrollar un programa en lenguaje ensamblador MIPS para extraer el valor de los tres números, y almacenar la extracción en el espacio reservado a partir de la etiqueta "**valores**".

Ejemplo: si en **mensaje** se encuentra el texto "los 2 sistemas contienen 9 archivos y 3 directorios", el programa debe extraer los valores 2, 9 y 3; y almacenar cada uno (su valor numérico, no su representación texto) en el espacio reservado en "**valores**". Tenga en cuenta que el mensaje puede variar, y por supuesto, los números en el mensaje también. Los números que existan en mensaje tienen un rango [0, 9].

Utilice el siguiente código en ensamblador como referencia para realizar el ejercicio.

```
.data
memoria:
valores:
    .word 0, 0, 0
mensaje:
    .asciiz "El resultado de la suma de los 3 números es 7, y es múltiplo de 2"

.text
.globl main
.globl __start

__start:
main:
    # COMPLETAR

# retorna al SO
move $a0, $0
li $v0, 4001
syscall
```

**Ejercicio 2.** Complete la subrutina siguiendo la convención de llamadas a procedimiento.

# esta subrutina obtiene un argumento N y devuelve 0 si el argumento N es par y 1 si el argumento N es impar.

es\_impar:

```
move $s3, ..... # colocar en $s3 el valor del argumento N de entrada a la función

..... # ahora utilizaremos la instrucción "and" para saber si $s3 es par o impar

move $t3, $s3 # colocamos el resultado (si es par o impar) en $t3

addi $a0, $a0, 48
jal mostrar_por_pantalla

move ....., $t3 # colocamos en .... el valor a retornar a la función llamadora

jr ..... # COMPLETAR
```

