

CONCEPTOS DE ARQUITECTURA DE COMPUTADORAS
PRÁCTICA 5 - PUNTO FLOTANTE

Ejercicio 7

\$a0 es M (positivo), \$a1 es dirección de inicio de tabla, \$a2 es cantidad de elementos

```
contar:      dadd  $v0, $zero, $zero      ; Inicializo retorno en 0
bucle:      ld    $t1, 0($a1)             ; elemento de tabla
            daddi $a2, $a2, -1             ; decrementar cant
            slt   $t2, $a0, $a2           ; t2 = M < elemento
            daddi $a1, $a1, 8              ; actualizar direccion
            dadd  $v0, $v0, $t2           ; actualizar retorno
            bnez  $a2, bucle               ; repetir si cant > 0
            jr    $ra
```

Ejercicio 8

\$a0 y \$a1 son las direcciones de inicio de dos cadenas terminadas en '\0'

```
string_eq:   dadd  $v0, $zero, $zero      ; Inicializo retorno en 0
bucle:      lbu    $t1, 0($a0)             ; Caracter de 1er cadena
            lbu    $t2, 0($a1)             ; Caracter de 2da cadena
            bneq   $t1, $t2, terminar      ; Saltar si no coinciden
            daddi  $v0, $v0, 1              ; Incremento retorno
            daddi  $a0, $a0, 1              ; Avanzar en 1er cadena
            daddi  $a1, $a1, 1              ; Avanzar en 2da cadena
            bnez   $t1, bucle               ; Repetir mientras no '\0'
            daddi  $v0, $zero, -1          ; -1 si son iguales
```

```
terminar:   jr    $ra
```

Ejercicio 9.1

```
factorial:   slti   $v0, $a0, 2            ; Posible caso base (n < 2)
            bnez   $v0, terminar           ; Saltar si dio true
            daddi  $sp, $sp, -8            ; Reservar espacio en pila
            sd     $ra, 0($sp)             ; Return address apilado
            daddi  $a0, $a0, -1            ; Nuevo argumento (n-1)
            jal    factorial               ; Llamado recursivo
            daddi  $a0, $a0, 1              ; Restaurar n (sin usar pila)
            dmul   $v0, $v0, $a0           ; Retorno = n * factorial(n-1)
            ld     $ra, 0($sp)             ; Return address desapilado
            daddi  $sp, $sp, 8              ; Stack pointer actualizado
terminar:    jr    $ra
```

Ejercicio 9.2

```
factorial:   daddi  $v0, $zero, 1          ; Inicializo retorno en 1
            beqz   $a0, terminar           ; Saltar si n = 0
bucle:      dmul   $v0, $v0, $a0           ; Actualizar retorno
            daddi  $a0, $a0, -1            ; Decrementar n
            bnez   $a0, bucle               ; Repetir si n > 0

terminar:    jr    $ra
```