

# Informe Tarea 4

## ARM Assembly

### Grupo 38

Diego Acevedo Pizarro 202023528-7  
Diego Paz Letelier 202004502-k

7 de diciembre de 2022

### Resumen

Según lo solicitado en la tarea se nos solicita escribir código en ARM Assembly que cumpla ciertas funciones, la primera es que logre calcular un determinante, la segunda que logre verificar si un par de palabras son anagramas, y la última es que calcule la distancia entre un punto inicial y otro final dado cuando recorre en cada eje, y a continuación mostraremos que hicimos para hacer esto

## Índice

1. Desarrollo de la tarea	1
2. Resultados y análisis	4
3. Conclusiones	6

## 1. Desarrollo de la tarea

Para la funcion 1 hicimos sumas y multiplicaciones simples accediento a los numeros mediante offsets en el registro en el que está almacenado el array de números

Para la funcion 3 hicimos que mientras compara el indice, va sumando a dos registros distintos de manera intercalada la distancia que se mueve en cada eje, para al final elevar ambas cantidades al cuadrado y sacar la distancia mediante raiz

Los codigos de ambas funciones se encuentran a continuacion en las páginas siguientes:

```

func1:
    mov r0,#0
    mov r1,#0
    mov r2,#0
    mov r3,#0
    ldr r0,=nums

    ldr r4,[r0,#0]
    ldr r5,[r0,#16]
    ldr r6,[r0,#32]
    mul r4,r5
    mul r4,r6
    mov r1,r4

    ldr r4,[r0,#4]
    ldr r5,[r0,#20]
    ldr r6,[r0,#24]
    mul r4,r5
    mul r4,r6
    mov r2,r4

    ldr r4,[r0,#8]
    ldr r5,[r0,#12]
    ldr r6,[r0,#28]
    mul r4,r5
    mul r4,r6
    mov r3,r4

    add r7,r1,r2
    add r7,r7,r3

    ldr r4,[r0,#0]
    ldr r5,[r0,#20]
    ldr r6,[r0,#28]
    mul r4,r5
    mul r4,r6
    mov r1,r4

    ldr r4,[r0,#4]
    ldr r5,[r0,#12]
    ldr r6,[r0,#32]
    mul r4,r5
    mul r4,r6
    mov r2,r4

    ldr r4,[r0,#8]
    ldr r5,[r0,#16]
    ldr r6,[r0,#24]
    mul r4,r5
    mul r4,r6
    mov r3,r4

    add r6,r1,r2
    add r6,r6,r3
    sub r6,r7,r6

    mov r0,#0
    mov r1,#0
    mov r2,r6
    bl printInt

```

Figura 1: Extracto de código

```

func3:
    ldr r0,=nums
mov r5,#0
mov r6,#0
b while
while:
    ldr r1,[r0,r3]
    ldr r2,[r0,r4]
    add r5,r5,r1
    add r6,r6,r2
    cmp r1,#0
    cmp r2,#0
    beq done

    add r3,r3,#8
    add r4,r4,#8
    b while
done:
mul r5,r5
mul r6,r6
add r5,r5,r6
mov r0,#1
mov r1,#1
mov r2,#0
b for
for:
    cmp r2,r5
    bgt done2

    mul r1,r1
    mov r2,r1
    add r0,r0,#1
    mov r1,r0
    b for

done2:
sub r0,r0,#2
mov r2,r0
mov r0,#0
mov r1,#2
bl printInt

```

Figura 2: Extracto de código

## 2. Resultados y análisis

Para testear nuestro código en Assembly hicimos uso de los ejemplos entregados en el pdf de la tarea, y a continuación están los resultados:

Función 1:

```
1 .data
2 func: .word 1
3 len1: .word
4 len2: .word
5 nums: .word 3,1,4,1,5,9,2,6,5
6 str1: .word
7 str2: .word
8 .text
9 main:
```

Figura 3: Parametros iniciales

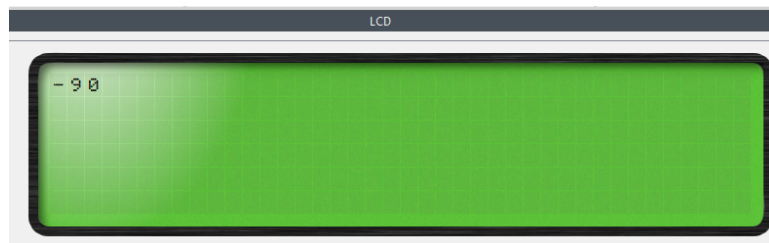


Figura 4: Resultado

```
1 .data
2 func: .word 1
3 len1: .word
4 len2: .word
5 nums: .word 2,7,1,2,8,2,1,2,8
6 str1: .word
7 str2: .word
8 .text
9 main:
```

Figura 5: Parametros iniciales

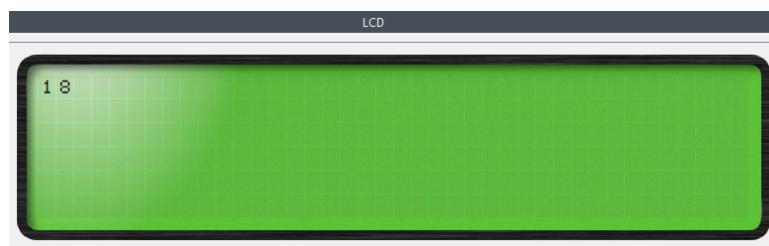


Figura 6: Resultado

Función 3:

```
1 .data
2 func: .word 3
3 len1: .word 7
4 len2: .word
5 nums: .word 0,2,7,1,-8,-2,-8
6 str1: .word
7 str2: .word
8 .text
9 main:
```

Figura 7: Parametros iniciales

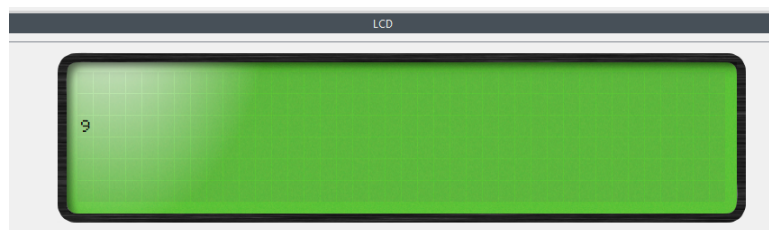


Figura 8: Resultado

```
1 .data
2 func: .word 3
3 len1: .word 6
4 len2: .word
5 nums: .word 1,2,3,-1,-2,-3
6 str1: .word
7 str2: .word
8 .text
9 main:
```

Figura 9: Parametros iniciales

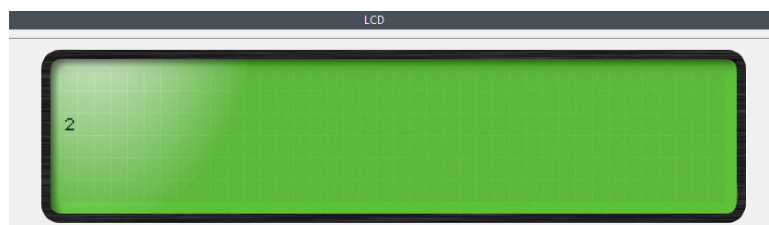


Figura 10: Resultado

Estos resultados coinciden con aquellos entregados en el pdf a excepcion de primero, pero en este caso es porque el ejemplo del pdf tiene un valor incorrecto y se puede ver puesto que es el mismo que esta en el ejemplo del inicio.

### **3. Conclusiones**

Por lo que podemos ver nuestro código en ARM Assembly es funcional y cumple con lo requerido para las funciones desarrolladas, el único detalle es que se nos complicó el desarrollo de la función 2 por temas de tiempo y coordinación por lo que no lo incluimos en este informe,