

---

## Laboratorio 4: Código ensamblador: ARMv4

---

Fecha de asignación: 24 abril 2024  
Grupos: 3 personas

Fecha de entrega: 3 mayo 2024  
Profesor: Luis Barboza Artavia

---

### 1. Introducción

El código ensamblador es un lenguaje de programación de bajo nivel que representa instrucciones específicas para un procesador de computadora. A diferencia de los lenguajes de programación de alto nivel, el código ensamblador está estrechamente vinculado a la arquitectura del procesador y utiliza mnemónicos y códigos de operación para representar las operaciones que la CPU debe realizar. Cada instrucción en código ensamblador se traduce directamente a una instrucción de máquina ejecutable por el procesador.

ARMv4 es una arquitectura de procesador desarrollada por ARM Holdings a principios de la década de 1990, conocida por su eficiencia energética y versatilidad. Introdujo características importantes como Thumb, una tecnología de instrucción de 16 bits para mejorar la densidad de código y el rendimiento en dispositivos con recursos limitados. ARMv4 fue ampliamente adoptada en una variedad de dispositivos móviles y embebidos, sentando las bases para el éxito continuo de ARM en la industria de semiconductores.

En este laboratorio el estudiante resolverá problemas mediante el uso de código ensamblador de ARMv4.

### 2. Investigación

Para el desarrollo de este laboratorio se deben responder las siguientes preguntas.

1. Investigue los tipos de instrucción que tiene ARMv4. Nombre dos ejemplos de cada tipo y un posible uso.
2. Explique el set de registros que tiene ARM. ¿Cuál es el contenido de cada uno?
3. Explique el funcionamiento del *branch*.
4. ¿Cómo se implementa un condicional (*if/else*) en ARMv4?
5. Explique el procedimiento para transformar el código en ensamblador a binario. Investigue herramientas que realizan el proceso.

- 
6. Explique la diferencia de *little endian* y *big endian*. ¿Cuál es su implicación cuando pasan las instrucciones a binario?

### 3. Ejercicios Prácticos

A continuación se presentan 3 ejercicios prácticos, los cuales debe resolver de manera completa. Para cada ejercicio debe presentar el código ensamblador en ARMv4, prueba del código ensamblador y el archivo con el binario equivalente.

#### 3.1. Problema 1

Asuma que se tiene un arreglo con 10 valores y una constante definida por el grupo. Realice el código ensamblador (ARMv4) para el siguiente pseudocódigo:

```
int [] array = {...}
int y =

for (i = 0, 1, ..., 10)
    if array[i] >= y
        array[i] = array[i] * y
    else
        array[i] = array[i] + y
```

#### 3.2. Problema 2

Realice el código ensamblador (ARMv4) para calcular Fibonacci hasta  $X$  número. Este número  $X$  irá “quemado” en el código. Realice la demostración para 3 números distintos.

#### 3.3. Problema 3

Realice el código ensamblador (ARMv4) para sumar dos matrices de números enteros de tamaño 4x4.

### 4. Metodología

Se utilizará la metodología explicada en el laboratorio 1.

Si tienen dudas puede escribir al profesor al [correo electrónico](#). **Los documentos serán sometidos a control de plagios.** La entrega se debe realizar por medio del TEC-Digital en la pestaña de evaluación. No se aceptan entregas extemporáneas después de la fecha de entrega a las 11:59 pm como máximo.