

# Informe N°1 Arquitectura de Computadores

Joaquín Saldivia (3.0 horas)  
*Departamento de Ingeniería Informática*  
*Universidad de Santiago de Chile, Santiago, Chile*  
 joaquin.saldivia@usach.cl

**Resumen**—Este trabajo presentara la solución diseñada para la falta de una biblioteca de operaciones vectoriales, utilizando el lenguaje MIPS, trabajando los datos por memoria y leyendo datos que necesitan ser trabajados, la solución se compone de operaciones matematicas vectoriales donde se encuentra la suma de vectores, multiplicacion por escalar y producto punto, logrando implementar todo de manera correcta y funcional.

**Palabras claves**—Memoria, Vectores y MIPS.

## I. INTRODUCCIÓN

El presente informe presenta una solución a la falta de una biblioteca para manipular vectores, donde se necesita idear en MIPS las operaciones de suma, producto por escalar y producto punto. Los objetivos son obtener mejor entendimiento respecto a MARS, implementación de algoritmos para resolver algoritmos matemáticos y escribir programas que utilizen instrucciones aritmeticas, de salto y de acceso a memoria, en la solución estara compuesto principalmente de procedimientos propios donde se aplican operaciones matematicas para vectores.

## II. ANTECEDENTES

### II-A. Algoritmos

#### Algoritmo 1 Suma de vectores

```
1: Input: Dos vectores, v1 y v2 de 3 componentes.
2: Output: Nulo.
3:  $i = 0$ 
4: print("[ ")
5: while  $i < 2$  do
6:    $temp = v1[i] + v2[i]$ 
7:   print(temp, )
8: end while
9: print("]")
10: return
```

#### Algoritmo 2 Multiplicacion por escalar

```
1: Input: Un vector v1 de 3 componentes y un escalar s.
2: Output: Nulo.
3:  $i = 0$ 
4: print("[ ")
5: while  $i < 2$  do
6:    $temp = v1[i] * s$ 
7:   print(temp, )
8: end while
9: print("]")
10: return
```

#### Algoritmo 3 Producto punto

```
1: Input: Dos vectores v1 y v2 de 3 componentes.
2: Output: Nulo.
3:  $i = 0$ 
4: while  $i < 2$  do
5:    $temp = temp + (v1[i] * v2[i])$ 
6: end while
7: print(temp)
8: return
```

### II-B. Fórmulas

Para la suma de vectores, asumiendo que existe los vectores  $[x_1, x_2, x_3]$  y  $[a_1, a_2, a_3]$  se tiene:

$$r_i = x_i + a_i \quad (1)$$

donde  $r_i$  corresponde a las componentes del vector resultante y  $x_i, a_i$  a las componentes de los vectores que se suman con  $i$  de 1 a 3.

Luego en la multiplicación por escalar, asumiendo que existe el vector  $[x, y, z]$  un escalar  $s$  se tiene:

$$[a, b, c] = [s \cdot x, s \cdot y, s \cdot z] = s * [x, y, z] \quad (2)$$

donde  $[a, b, c]$  corresponde al vector resultante.

Finalmente para el producto punto, teniendo los vectores  $[x_1, x_2, x_3]$  y  $[a_1, a_2, a_3]$  se tiene:

$$[x_1, x_2, x_3] \cdot [a_1, a_2, a_3] \rightarrow r = \sum x_i * a_i \quad (3)$$

donde  $r \in \mathbb{Z}$  e  $i$  va de 1 a 3 .

## III. MATERIALES Y MÉTODOS

Como su nombre lo indica, esta sección se descompone en dos subsecciones: materiales y métodos.

### III-A. Materiales

Se requiere de un computador para utilizar el lenguaje MIPS junto con el IDE MARS para poder editar y ejecutar de manera correcta este lenguaje, tambien se requiere de tener memoria disponible para realizar el trabajo.

### III-B. Métodos

Describe el procedimiento desarrollado. Puede contener subsecciones.

El metodo que mas se ocupará es el de acceso y manipulación de registros, en el que se encuentran 3 tipos[1]:

- Tipo R: Instrucciones de Registro.
- Tipo I: Instrucciones Inmediatas.
- Tipo J: Instrucciones de Salto.

Donde en conjunto de estos se diseña una solución para el problema 1, donde se necesita realizar una suma de vectores, que consiste en realizar un ciclo donde se recorre desde la memoria 0x10010000 hasta 0x10010008 para el vector 1 y desde la memoria 0x10010020 hasta 0x10010028 para el vector 2 y en este recorrido se realiza una suma de entre componentes y se imprime repitiendo esto hasta operar con los 3 valores.

Luego para la solución para el problema 2, donde se necesita realizar una multiplicación por escalar, que consiste en realizar un ciclo donde se recorre desde la memoria 0x10010040 hasta 0x10010048 para el vector 1 y al recorrerlo se realiza una multiplicación entre el componente del vector y el escalar, esto se imprime repitiendo esto hasta operar con los 3 valores..

Finalmente la solución para el problema 3, donde se necesita realizar un producto punto, que consiste en realizar un ciclo donde se recorre el vector 1 obtenido por consola, desde una etiqueta asignada, de espacio 12 y desde la memoria 0x10010060 hasta 0x10010068 para el vector 2, se toma un acumulador que durante cada iteración suma la multiplicación entre componentes hasta que realiza estos 3 ciclos para terminar imprimiendolo en la consola.

## IV. RESULTADOS

Se obtienen los resultados esperados, donde cada operacion funciona como deberia y esto se puede observar en las siguientes tablas:

Tabla I: Resultados Suma de Vectores

	Vector 1	Vector 2	Vector Obtenido
Ejemplo 1	[2, 2, 2]	[5, 30, 49]	[7, 32, 51]
Ejemplo 2	[13, 7, 10]	[42, 12, 4]	[55, 19, 14]
Ejemplo 3	[1, 2, 3]	[3, 2, 1]	[4, 4, 4]
Ejemplo 4	[22, 21, 24]	[12, 42, 22]	[34, 63, 46]

Tabla II: Resultados Multiplicación por Escalar

	Vector 1	Escalar	Vector Obtenido
Ejemplo 1	[2, 2, 2]	3	[6, 6, 6]
Ejemplo 2	[13, 7, 10]	2	[26, 14, 20]
Ejemplo 3	[1, 2, 3]	15	[15, 30, 45]
Ejemplo 4	[5, 10, 15]	4	[20, 40, 60]

## V. CONCLUSIONES

Se logran cumplir los objetivos, teniendo así un mejor entendimiento del funcionamiento de registros y memorias a

Tabla III: Resultados Producto Punto

	Vector 1	Vector 2	Valor Obtenido
Ejemplo 1	[2, 2, 2]	[1, 1, 1]	6
Ejemplo 2	[5, 7, 2]	[2, 1, 4]	25
Ejemplo 3	[1, 2, 3]	[3, 2, 1]	10
Ejemplo 4	[10, 20, 30]	[4, 3, 7]	310

nivel de computador a través del lenguaje MIPS, empleando correctamente el movimiento secuencial de estas ultimas, asi como tambien un aplicación que resuelve correctamente el problema planteado inicialmente, contribuyendo asi a esta nave con el modulo que le permite navegar por el oceano.

## REFERENCIAS

- [1] C. T.-G. y. R. A.-I. A. Hernández-Cerezo, "Arquitectura mips," in *Asignatura de Organización de Computadoras*, vol. 1, no. 2, pp. 2–15.