



Laboratorio 4. Acercándose al Hardware: Programación en Lenguaje Ensamblador

Objetivos de aprendizaje

- Usar MARS (un simulador para MIPS) para escribir, ensamblar y depurar programas MIPS
- Escribir programas MIPS incluyendo instrucciones aritméticas, de salto y memoria
- Comprender el uso de subrutinas en MIPS, incluyendo el manejo del *stack*
- Realizar llamadas de sistema en MIPS mediante “*syscall*”
- Implementar algoritmos en MIPS para resolver problemas matemáticos

Entrega

Sube los archivos creados, junto con tu informe, a través de Google Classroom. **Todos los archivos con código MIPS deben poder ensamblarse y ejecutarse en el simulador MARS, y deben estar debidamente comentados.** Guarda cada programa en un archivo distinto con el nombre de la parte que corresponde. Por ejemplo, tu trabajo de la Parte 1, pregunta A, guárdalo en el archivo “parte1a.asm”.

Para todos los programas se debe pedir por consola el número a ingresar y se debe mostrar el resultado por consola mediante los llamados *syscall*.

Parte 1: Aproximación de funciones matemáticas

Se pide escribir un programa que calcule una aproximación de las siguientes funciones, evaluadas en un **número entero no negativo**:

- 1) Exponencial (2.0 puntos)
- 2) Coseno (2.0 puntos)

Para aproximar estas funciones, utiliza expansiones de Taylor en torno a 0, de orden 7 o superior. El programa **debe** utilizar para el cálculo de las multiplicaciones, divisiones y factorial los procedimientos del laboratorio anterior (recordar instrucciones prohibidas). Además, el programa debe pedir al usuario, via consola MARS, el número entero no negativo para el cual se quiere evaluar la función, y debe mostrar en la misma consola los resultados con 2 decimales, sin atender a errores de precisión.

Función exponencial:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$



Función coseno:

$$\cos x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} \quad \text{para toda } x$$

La frase “en torno a 0 y de orden 7 o superior” hace referencia a que la sumatoria debe empezar desde 0 (valor “n” que se observa en las imágenes) y debe llegar hasta orden 7 o superior. En la parte de resultados del informe se debe responder a lo siguiente: ¿Qué efecto tiene aumentar el orden de la sumatoria en los resultados?

Parte 2: Implementación en MIPS de una función recursiva (2.0 puntos)

La sucesión de Fibonacci comienza con los números 0 y 1 y, a partir de estos, cada término se calcula como la suma de los dos anteriores. Considera la siguiente implementación en lenguaje de alto nivel de una función recursiva que entrega el elemento n-ésimo de la sucesión de Fibonacci:

```
int fibonacci(int n) {  
    if (n == 0) {  
        return 0;  
    } else if (n == 1) {  
        return 1;  
    }  
    return fibonacci(n-1) + fibonacci(n-2);  
}
```

Escribe un programa en MIPS que calcule el elemento n-ésimo de la sucesión de Fibonacci según el esquema recursivo de la función mostrada arriba. Como condición, se pide no convertir a una solución iterativa.

Informe

El informe para entregar debe contar con lo siguiente:

- Introducción que incluya el problema, solución y objetivos de esta experiencia (0.5 puntos)
- Marco teórico que explique los conceptos necesarios para entender el trabajo desarrollado (0.5 puntos)
- Explicación breve del desarrollo de la solución y cómo se llegó a esta (2.0 puntos)
- Resultados de cada parte del laboratorio (2.0 puntos)
- Conclusiones (0.7 puntos)
- Referencias bibliográficas. Debe citar todas las fuentes externas usadas. Formato APA. (0.3 puntos)



Exigencias

- El informe escrito debe ser entregado en formato PDF. Se sugiere una extensión no mayor a 10 páginas. Tanto el código fuente como el informe deben ser enviados por medio de la plataforma Google Classroom **en un archivo comprimido .zip**, cuyo nombre debe incluir el RUT de el/la alumno/a (ej: lab4_12345678-9.zip).

Descuentos

- Entregas que no cumplan con el formato indicado se descontarán 2.0 puntos.
- Por cada tres faltas ortográficas o gramaticales en el informe, se descontará una décima a la nota del informe.
- En caso de que no se entregue el informe o el código fuente, se evaluará con la nota mínima.
- **Esta permitido realizar consultas a chatgpt sobre funciones, instrucciones, ..., sin embargo, si el código es similar a la respuesta que da chatgpt al colocar la pregunta en el prompt, su respuesta quedará invalidada y con 0 puntos.**
- La nota del laboratorio será calculada como: $0.7 \cdot \text{código} + 0.3 \cdot \text{informe}$.
- **Fecha entrega diurno:** Lunes 30 de junio de 2025 a las 23:59 hrs.
- **Fecha entrega vespertino:** Lunes 07 de julio de 2025 a las 23:59 hrs.