

## PRÁCTICA DE LABORATORIO 4

---

### OBJETIVO

Familiarizar al estudiante con la sintaxis del lenguaje ensamblador MIPS32 y su ejecución en el entorno MARS, resolviendo problemas clásicos de programación, prestando especial atención al uso de interrupciones para la gestión de **entrada/salida (E/S)** y la utilización de rutinas de servicio para responder a eventos como **entrada por teclado y control de tiempo**, simulando el funcionamiento de periféricos.

---

### EJERCICIOS

1. Buffer de Entrada de Teclado con Temporizador (Interrupciones simuladas). Desarrollar un programa en MIPS32 que funcione de la siguiente manera:

Durante un intervalo de **20 segundos**, el programa debe permitir la **entrada de caracteres por teclado**, que se irán almacenando en un **buffer circular**.

Transcurridos los 20 segundos, el programa **imprime todo el contenido del buffer** en la consola.

Después de vaciar el buffer, el proceso **se repite indefinidamente**.

2. Semáforo con Pulsador (Interrupciones de Teclado y Temporizador). Desarrollar un programa en MIPS32 que simule un **semáforo controlado por un pulsador (tecla s)** con el siguiente funcionamiento:

El semáforo comienza en **verde** mostrando: **“Semáforo en verde, esperando pulsador”**.

Al pulsar la tecla **s**, el semáforo debe mostrar: **“Pulsador activado: en 20 segundos, el semáforo cambiará a amarillo”**. Iniciar un temporizador de 20 segundos.

Tras los 20 segundos, el semáforo pasa a **amarillo** mostrando: **“Semáforo en amarillo, en 10 segundos, semáforo en rojo”**.

Pasados 10 segundos, pasa a **rojo** mostrando: **“Semáforo en rojo, en 30 segundos, semáforo en verde”**

Después de 30 segundos, vuelve a **verde** y se repite el ciclo.

---

### INFORME

1. Explique con un diagrama **cómo funciona el ciclo de una interrupción de hardware** (desde que ocurre el evento hasta que se atiende).
2. ¿Qué diferencias hay entre gestionar E/S con sondeo y hacerlo con interrupciones?
3. ¿Qué ventajas tiene el uso de interrupciones en términos de uso del procesador?
4. ¿Qué registros especiales se utilizan en MIPS32 para gestionar interrupciones?

5. ¿Por qué es necesario guardar el contexto (registros) al entrar en una rutina de servicio?
6. Momentos en que pueden generarse excepciones en un sistema MIPS32.
  - a) Enumera al menos 4 situaciones en las que se pueda generar una excepción (por ejemplo: desbordamiento aritmético, fallo de dirección, etc.).
  - b) Explica qué etapas del pipeline pueden provocar una excepción y por qué.
7. Estrategias de tratamiento de excepciones e interrupciones
  - a) Explica las diferencias entre interrupciones y excepciones (¿son síncronas o asíncronas?).
  - b) Describe brevemente dos estrategias para tratar excepciones en un sistema MIPS32 ¿Cómo se redirige la ejecución hacia la rutina de servicio? ¿Cuál es la función del registro EPC (Exception Program Counter)?
8. Habilitación de interrupciones en dispositivos y procesador
  - a) Explica cómo se habilitan las interrupciones en:
    - El teclado.
    - La pantalla.
    - El procesador (detalla qué bits del registro Status deben modificarse).
  - b) ¿Qué pasaría si habilitamos interrupciones en los dispositivos, pero no en el procesador?
9. Procesamiento de interrupciones
  - a) Describe paso a paso qué ocurre cuando se produce una interrupción de reloj:
    - Desde que el evento ocurre hasta que la rutina de servicio termina.
    - ¿Qué registros se guardan y restauran?
    - ¿Qué hace el hardware y qué hace el software (sistema operativo o rutina)?
  - b) ¿Por qué es importante guardar el contexto (registros generales, EPC, Status) al entrar en la rutina?
10. Interrupciones de reloj y control de ejecución
  - a) Explica cómo una interrupción de reloj puede usarse para:
    - Evitar que un programa quede en un bucle infinito.
    - Finalizar programas que superan un tiempo máximo de ejecución.
  - b) ¿Qué debe hacer el software si el programa finaliza antes de que ocurra la interrupción de reloj?
11. Análisis y Discusión de los Resultados

---

## ENTREGA

1. Práctica de Laboratorio **Individual o en Pareja.**
2. Se deben enviar al correo los archivos .asm junto con el documento (.tex y .pdf) con las respuestas.
3. Se debe garantizar la correcta ejecución de los archivos .asm y visualización del documento (.tex y .pdf) con las respuestas en la fecha prevista para la defensa.
4. La defensa presencial de cada práctica de laboratorio es obligatoria.
5. La fecha máxima de entrega y fecha de defensa es el día martes 29 de julio de 2025 en las instalaciones del laboratorio.