

# Arquitectura de Computadores

## Laboratorio 2

1 de Junio, 2023

### Resultados de aprendizaje

1. Usar MARS (un *IDE* para MIPS) para escribir, ensamblar y depurar programas MIPS.
2. Escribir programas MIPS incluyendo instrucciones aritméticas, de salto y de acceso a memoria. Incluyendo el uso de subrutinas.
3. Realizar llamadas de sistema en MIPS mediante uso de `syscall` empleando procedimientos.
4. Implementar algoritmos en MIPS para resolver problemas matemáticos sencillos.

### MIPS del mas allá

Esteban Strange es un científico brillante, quien es conocido por su obsesión con el mundo de lo paranormal y lo desconocido. Siempre busca desafíos que pusieran a prueba su valentía y habilidades científicas. Hace unas semanas, mientras investigaba en su laboratorio, se encontró con un antiguo libro encuadernado en cuero oscuro, que estaba lleno de extraños símbolos y fórmulas matemáticas. Intrigado por el misterioso libro, Esteban comenzó a estudiarlo detenidamente y descubrió que los símbolos y las fórmulas eran una antigua técnica para invocar fuerzas sobrenaturales. Según los escritos, este conocimiento ancestral podía utilizarse para desbloquear los secretos del universo, incluido el enigma de  $\pi$ .

Movido por una extraña mezcla de curiosidad y temor, Esteban decidió adentrarse en este mundo paranormal y utilizar sus habilidades científicas para resolver el enigma de  $\pi$  de una manera sobrenatural. Recordando una leyenda que decía que  $\pi$  estaba oculto en un lugar llamado "La Dimensión Espectral", Esteban se dispuso a construir un portal que lo llevara a este reino desconocido. Utilizando su conocimiento en lenguaje ensamblador MIPS, diseñó un programa especial que, combinado con una serie de artefactos místicos, permitiría la apertura del portal hacia La Dimensión Espectral. Sabía que en ese lugar encontraría respuestas a preguntas que el mundo científico aún no podía resolver.

Siguiendo las instrucciones del libro, Esteban ejecutó su programa en el laboratorio a altas horas de la noche. El lugar se llenó de una energía inquietante y una neblina oscura se materializó frente a sus ojos. Temeroso pero decidido, Esteban cruzó el umbral y se encontró en un reino paralelo, lleno de figuras etéreas y sombras que se movían en formas desconcertantes.

A medida que se adentraba más en La Dimensión Espectral, Esteban notó que las figuras y las sombras formaban patrones que se asemejaban a los números y las fórmulas matemáticas que había estudiado. Se dio cuenta de que este reino era una manifestación física de los conceptos abstractos de las matemáticas, incluido el enigma de  $\pi$ .

Fue en ese momento en el que se dio cuenta que para poder resolver el enigma, y con ello poder volver a su dimensión, debía crear una programa en MIPS que aproximara el valor de  $\pi$  por medio del método de Monte Carlo. Sin embargo, su computador estaba en la otra dimensión y en ese momento no había nadie más en su laboratorio.

Luego de un par de horas, un ayudante de su laboratorio encontró todo el desastre que se había generado. Vio en el laboratorio el computador de Esteban y el portal. Del otro lado del éste se encontraba el científico el cual apenas ve a su ayudante intenta comunicarle lo ocurrido. Ambos se pueden oír, pero Esteban no puede cruzar el portal de vuelta. Es por ello que le pide a su ayudante que escriba el programa en MIPS que le permitiría salir de La Dimensión Espectral. Sin embargo, su ayudante no sabía programar en dicho lenguaje ensamblador.

Desesperado por ayudar lo más pronto posible al científico Esteban, su ayudante solicita ayuda a los estudiantes del curso de Arquitectura de Computadores para que puedan desarrollar el código MIPS que permita traer de vuelta a su tutor. Para ello el ayudante les dice palabra por palabra lo que le había dicho el científico Esteban:

''Se debe aproximar el número  $\pi$  utilizando el método Monte Carlo. Para ello debes generar puntos aleatorios en formas etéreas para luego calcular distancias y determinar si los puntos se encuentran dentro del círculo sobrenatural. Todos esos pasos se deben realizar utilizando subrutinas místicas... A medida que se agreguen más puntos aleatorios la energía de esta dimensión se volverá más intensa, lo cual indica que nos acercamos el valor de  $\pi$  que armoniza con la esencia de La Dimensión Espectral... Al obtener una aproximación de  $\pi$  resonante, debería poder cruzar este portal de vuelta...''

## Aproximación de $\pi$ por método de Monte Carlo

El método de Monte Carlo es una técnica utilizada para aproximar el valor de  $\pi$  utilizando principios estadísticos y aleatorios. La idea principal detrás del método es generar un gran número de puntos aleatorios dentro de un cuadrado unitario y determinar cuántos de estos puntos están dentro de un círculo unitario inscrito en el cuadrado. Al comparar la proporción de puntos dentro del círculo con la proporción de puntos totales generados, podemos obtener una estimación de  $\pi$ .

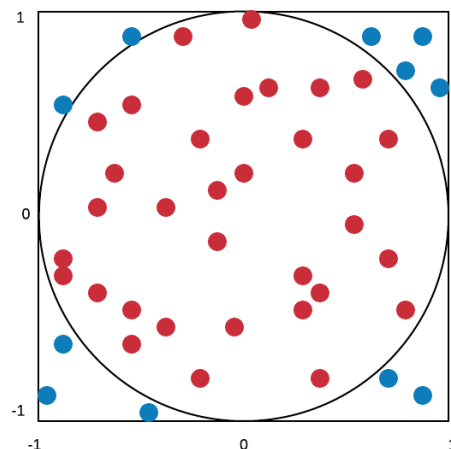


Figure 1: Representación gráfica de método de Monte Carlo para aproximación de  $\pi$

Para aplicar el método de Monte Carlo para aproximar pi, primero se debe generar un gran número de puntos aleatorios con coordenadas (x, y) dentro del rango [0, 1] en ambos ejes. Luego, se calcula la distancia euclidiana desde el origen (0, 0) hasta cada punto generado. Si la distancia es menor o igual a 1, el punto se encuentra dentro del círculo unitario. Luego, se cuenta el número de puntos que están dentro del círculo y dividimos este valor por el número total de puntos generados. La proporción obtenida se puede multiplicar por 4 para obtener una aproximación de pi, ya que el área del círculo unitario es  $\pi/4$  en relación con el área del cuadrado unitario. Es decir:

$$\pi = 4 * \frac{\text{puntos en el círculo}}{\text{puntos en total}} \quad (1)$$

## Implementación

Teniendo en consideración el método de Monte Carlo y lo solicitado por el científico Esteban, el código a desarrollar debe ser implementado utilizando subrutinas, en específico se deben implementar tres obligatoriamente (con el mismo nombre que se indica a continuación):

- **GenerarPuntoAleatorio:** Esta subrutina debe generar un par de coordenadas (x, y) aleatorias entre 0 y 100 (se deben incluir los extremos).
- **CalcularDistancia:** Esta subrutina debe calcular la distancia entre el punto (x, y) y el origen de coordenadas, utilizando la fórmula de la distancia euclidiana.
- **PuntoDentroDelCirculo:** Esta subrutina debe recibir las coordenadas (x, y) de un punto y determinar si se encuentra dentro del círculo de radio 1 unidad y centro en el origen. Debe retornar un valor booleano (1 si está dentro, 0 si está fuera).

Para poder generar valores aleatorios en MIPS se debe hacer uso de un `syscall`, el cual se puede ver en el siguiente código de ejemplo:

```
.data
    random_msg: .ascii "Número aleatorio entre 0 y 5 generado: "

.text

main:
    # Se imprime mensaje de numero aleatorio
    la $a0, random_msg
    li $v0, 4
    syscall

    # Generar número aleatorio entre 0 y 5
    li $a1, 6
    li $v0, 42
    syscall

    # Se imprime por consola el número generado
    li $v0, 1
    syscall
```

Figure 2: Código ejemplo de como generar un número aleatorio en MIPS

Dado que en MIPS no hay una forma nativa de generar números aleatorios que sigan una distribución uniforme, se utilizará una versión modificada del método Monte Carlo. Normalmente se utiliza un círculo unitario (valores entre 0 y 1), sin embargo, para efectos de esta experiencia se utilizará un círculo de radio 100. En términos prácticos la fórmula a utilizar es la misma (Ecuación 1).

## Entrega

El/la estudiante deberá subir el código desarrollado, junto con su informe, a través de la plataforma Campus Virtual del curso. El archivo con código MIPS Assembly debe poder ensamblarse y ejecutarse en el simulador MARS, y deben estar debidamente comentados. El código debe ser entregado con el siguiente nombre `codigo_NombreApellido.asm`.

## Informe

El informe a entregar debe contar con lo siguiente:

1. Introducción que incluya el problema, solución y objetivos de esta experiencia.
2. Marco teórico que explique los conceptos necesarios para entender el trabajo desarrollado.
3. Explicación breve del desarrollo de la solución y cómo se llegó a esta.
4. Resultados del laboratorio.
5. Conclusiones.

El detalle de la rúbrica para la evaluación de esta experiencia estará disponible en la sección respectiva de Moodle.

## Exigencias

1. El código MIPS a realizar, debe se debe crear a partir de los conocimientos presentados en cátedra.
2. El código será ejecutado y probado en el simulador MARS. Por lo que, si el código escrito no funciona en dicho simulador, se considerará como nota mínima el código.
3. El informe escrito debe ser entregado en formato PDF y no puede exceder 5 páginas de contenido. En caso contrario, por cada página extra se descontará 5 décimas a la nota final.
4. El informe se debe entregar con el siguiente formato de nombre `informe.NombreApellido.pdf`
5. El informe se debe desarrollar en base a la plantilla que se dejará disponible en Campus Virtual.
6. Tanto el código fuente como el informe deben ser enviados en un archivo comprimido, cuyo nombre debe incluir el RUT de el/la alumno/a (ej: lab1\_12345678-9.zip)

## Recomendaciones

1. Consultar al ayudante en las sesiones de laboratorio.

## Descuentos

1. Por cada día de atraso se descontará un punto a la nota final del laboratorio.
2. Por cada tres faltas ortográficas o gramaticales en el informe, se descontará una décima a la nota del informe.
3. Por cada falta de formato en el informe se descontará una décima a la nota del informe.
4. Por cada página extra en el informe se descontarán cinco décimas a la nota final del informe.

## Evaluación

1. La nota del laboratorio será el promedio aritmético entre la nota del informe y la nota del código fuente.
2. En caso de que no se entregue el informe y/o el laboratorio, se evaluará con la nota mínima.
3. Este laboratorio debe ser entregado el **15 de Junio** de 2023, a las 23:59 hrs por Campus Virtual.