

# Trabajo Práctico 3

## Sincronización de procesos

Tecnología Digital II

El trabajo práctico debe realizarse en grupos de tres personas. Tienen una semana para realizarlo. **La fecha de entrega límite es el domingo 14 de noviembre hasta las 23:59.** Se solicita no realizar consultas del trabajo práctico por los foros públicos. Limitar las preguntas al foro privado creado para tal fin.

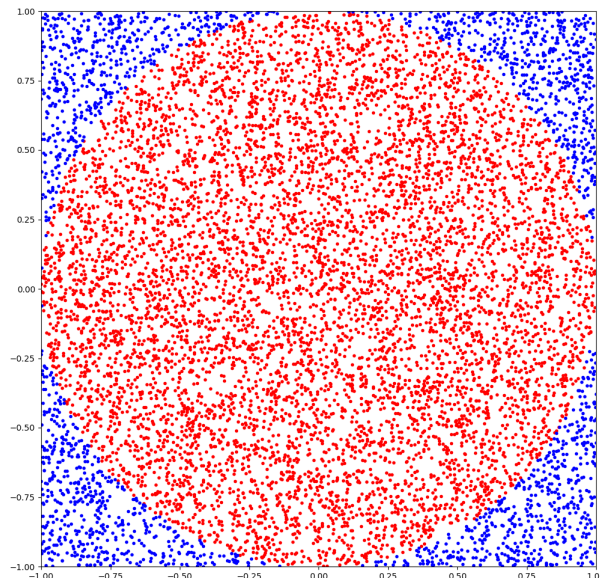
### Enunciado

El objetivo de este Trabajo Práctico es estimar el valor de  $\pi$  mediante un método estadístico, el mismo que utilizamos en TD1 para resolver el ejercicio 6 de la Guía 9. La estimación se basa en una metodología general de aproximación de expresiones matemáticas complejas o valores no racionales conocida como **Método de Montecarlo**.

Para estimar  $\pi$  se inscribe un círculo de radio 1 en un cuadrado de lado 2. El área del círculo es  $\pi \cdot \text{radio}^2 = \pi$ . El área del cuadrado es  $2 \cdot 2 = 4$ . Por lo tanto, dividir el área del círculo con la del cuadrado da como resultado  $\frac{\pi}{4}$ . Como no conocemos  $\pi$ , lo que hacemos es estimar la proporción entre las áreas del círculo y del cuadrado, y luego inferir  $\pi$ . Para ello:

1. Elegimos N puntos de manera aleatoria y uniformemente distribuidos dentro del cuadrado. Algunos estarán dentro del círculo mientras que otros estarán fuera de él.
2. Contabilizamos los puntos que quedaron dentro del círculo, es decir, aquellos cuya distancia al centro sea menor o igual a 1.
3. Dividimos la cantidad de puntos dentro del círculo con N, para hallar una aproximación de  $\frac{\pi}{4}$ .
4. Multiplicamos la estimación anterior por 4 para obtener la aproximación de  $\pi$ .

En el siguiente gráfico se puede apreciar el resultado de elegir N puntos al azar dentro un cuadrado de 2x2:



## Ejercicio 1

Se debe completar el código del programa `monteCarloPI_Monoproceso.c` en donde se calcula la estimación de  $\pi$  en un único proceso. En el código debe reemplazar todas las apariciones del texto **COMPLETAR** por una o varias instrucciones que terminen de implementar el programa.

**Estimación de las líneas de código faltantes:** menos de 5.

## Ejercicio 2

Se debe completar el código faltante del programa `monteCarloPI_Mutiproceso.c`. Este programa estima  $\pi$  utilizando exactamente 10 procesos a la vez. Cada proceso se encarga de generar y contar puntos al azar dentro del círculo y del cuadrado. Ambos valores son reportados al proceso padre que se ocupa de sumar los resultados y calcular la estimación. Como los procesos se tienen que comunicar entre sí, lo hacen mediante archivos. Cada proceso, cuando termina, escribe dos archivos con los datos que calculó. Luego, el proceso padre, lee estos archivos y realiza el cálculo para estimar  $\pi$ . Las funciones que utilizan para leer y escribir archivos son respectivamente `fileRead` y `fileWrite`. Tener en cuenta que las únicas instrucciones que están permitidas para crear y sincronizar los procesos son `fork()` y `wait()`.

**Estimación de las líneas de código faltantes:** menos de 20.

## Ejercicio 3 (Optativo)

El programa anterior está diseñado para construir como máximo 10 procesos. Esto se debe a la forma en como arma los nombres de los archivos donde escribir los resultados. Este ejercicio consiste en modificar el código anterior para soportar una cantidad arbitraria de procesos pasados por parámetro. Adicionalmente, se pide alterar el código para que se escriba un solo archivo por proceso. En la solución original, cada proceso escribe un archivo para la cantidad de puntos del cuadrado y otro para la cantidad de puntos del círculo. Para esto puede utilizar cualquier función de la `libc`, como `sprintf`, `atoi` o `scanf`.

## Interpretación del código

Los 2 programas incompletos que les damos incluyen instrucciones y tipos de datos que no vimos en clase. Parte del objetivo del trabajo es que investiguen de qué se trata cada elemento del código que no conozcan como, por ejemplo: `int64_t`, `srand`, `getpid()`, etc.

## Entregable

La entrega debe constar de los mismos 2 archivos `.c` que recibieron, pero con las partes faltantes completadas. El código que les damos ya hecho no debe ser alterado. Al correr cualquiera de los 2 programas, lo último que debe imprimir es el valor aproximado de  $\pi$ . Además, no deben quedar creados archivos luego de terminada la ejecución.

El código debe estar comentado, tanto las partes que ya les damos hechas como las que deben hacer ustedes. En ambos casos, deben explicar qué aspecto del problema resuelve cada parte del código y cómo la resuelve. No explicar instrucción por instrucción.