

P1.- Calcular Pi



UNIVERSIDAD
DE GRANADA



Nicolás Calvo Cruz
Dpto de Arquitectura y Tecnología de los Computadores
@ncalvocruz
ncalvocruz@ugr.es

Índice

1. Bases
2. Acceso a atcgrid.ugr.es
3. Prueba atcgrid.ugr.es
4. Ejercicios

Bases

- Las prácticas se realizan en Linux nativo
- No se hacen en Windows
- No se hacen en una Máquina Virtual con Linux
- Se usarán al menos dos máquinas: la personal o del aula y atcgrid.ugr.es
- Las prácticas se entregan en swad.ugr.es en la fecha indicada.
- Entregar tarde reduce la nota
(pero es mejor que no entregar)

Acceso a atcgrid.ugr.es

- ❖ Repasad el documento *AC_Seminario0_entorno.pdf*

Logins

- Recibir del profesor
- El login es personal e intransferible
- Las contraseñas NO se pueden cambiar

Procedimiento

- El procedimiento para acceder a atcgrid.ugr.es incluye
 - ssh
 - scp
login@atcgrid.ugr.es:/pathAFi
chero directorioLocal

Prueba atcgrid.ugr.es

1. Ver el material asignado a la práctica 1 en SWAD
2. Descargar el material en vuestra máquina local
3. Ejecutar en vuestra máquina local compilando con gcc
4. Subir código fuente a atcgrid.ugr.es con scp
5. Compilar código en atcgrid
6. Enviar a la cola de ejecución
7. Recuperar resultados

```
MBP-de-Nicolas:Desktop nccruz$ ./myPi 1000
PI por la serie de G. Leibniz [1000 iteraciones] =      3.140593
PI por integración de círculo [1000 intervalos] =      3.141593
```

Cálculo de Pi

- El número π es la relación entre la longitud de una circunferencia y su diámetro.
- Es un número irracional y una de las constantes más usada en Matemáticas, Física e Ingeniería.

$$\pi = 3.141592653589793238462... \approx 3.141593$$

- Su valor puede estimarse con diversas aproximaciones, como el [Algoritmo de Chudnovsky](#) y la [Serie de Leibniz](#) entre otras.
- El programa proporcionado en **pi.c** implementa dos de ellas...

Cálculo de Pi (versión 1)

- La primera de ellas es la **Serie de Leibniz**, que define la siguiente igualdad:

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots = \frac{\pi}{4}$$

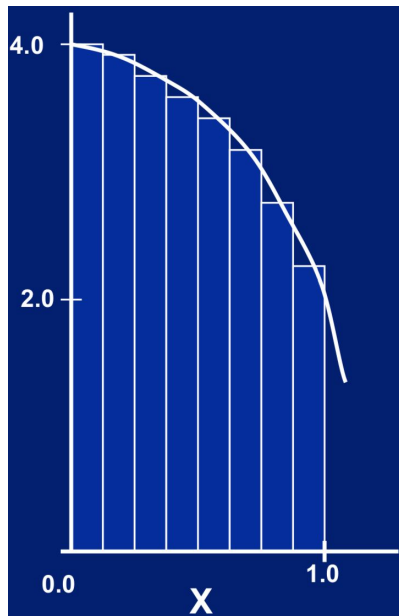
- Esta puede generalizarse así:

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = \frac{\pi}{4}$$

*Más detalles en Wikipedia: https://es.wikipedia.org/wiki/Serie_de_Leibniz

Cálculo de Pi (versión 2)

- La segunda de ellas aproxima el valor sabiendo que: $\int_0^1 \frac{4}{1+x^2} dx = \pi$



- A su vez, esa integral se puede aproximar como una suma del área de N rectángulos:

$$\sum_{i=0}^N F(x_i) \Delta x \cong \pi$$

- Donde cada rectángulo tiene una base fija Δx y una altura dada por $F(x) = 4/(1+x^2)$

Práctica a realizar (I)

- Inserta una medición de tiempo real de ejecución del programa de cálculo de Pi que incluya ambas aproximaciones (y sin tener en cuenta entrada/salida).
- En tu máquina, haz un estudio de cómo evoluciona el tiempo de ejecución del programa conforme se incrementa el número iteraciones hasta el máximo entero representable, con 10 pasos (¿[limits.h](#)?). No olvides realizar varias mediciones para cada caso para obtener su media y desviación típica.
- Dibuja una gráfica que muestre la evolución del tiempo medio de ejecución obtenido en cada experimento, comentando cómo es ésta.

Práctica a realizar (II)

- Prueba a ejecutar en `atcgrid.ugr.es` el caso más costoso utilizando un solo nodo de cómputo de la máquina, la cuenta `acap` y la cola `acap`:
 - `gcc fichero.c -o fichero.o`
 - `srun -Aacap -p acap ./pi <mayor entero>`
- Crea un script para lanzar las ejecuciones del algoritmo en `atcgrid`.
- Paraleliza los cálculos con MPI dividiendo las iteraciones entre el número de procesos disponibles. Busca el **máximo rendimiento** (*¿combinar?*).
- Haz ejecuciones para al menos cuatro configuraciones de procesos diferentes (Ej: 1, 2, 3, 4), tanto en tu máquina como en `atcgrid`.

Práctica a realizar (III)

- Realiza los cálculos necesarios para calcular la ganancia en velocidad que alcanza tu algoritmo paralelo:
 - En tu máquina local
 - En atcgrid.ugr.es (usando sbatch, <https://slurm.schedmd.com/sbatch.html>)
 - `sbatch -Aacp -p acp -N 1 -n 2 -c 1 --hint=nomultithread --exclusive --wrap "mpirun -np 2 ./programa"`
- Realiza los cálculos necesarios para calcular la eficiencia y represéntala para cada una de las configuraciones de procesos necesarias.
- Justifica tus decisiones y explica todos tus resultados.
- Las gráficas deben llevar toda la información en los ejes y en la leyenda para saber qué representa, en qué escala, unidades, etc.

—

**Tiempo de realización: 2
semanas**

**Fecha de entrega: 16 de
marzo**

Se entregan:

- 1. Ficheros con algoritmo secuencial y paralelo ya parametrizados que incluyan dentro vuestro nombre y la fecha**
- 2. Scripts de compilación y ejecución en máquina local y en atcgrid para versión paralela y secuencial con nombre y fecha**
- 3. Fichero en formato pdf con resultados y explicaciones (máx 3 páginas) con nombre y fecha**

iVamos a trabajar!