

Objetivo

El objetivo es estimar el valor de π , paralelizando la implementación de dos métodos:

1. El primero, que calcula el área bajo la curva del primer cuadrante de un círculo mediante integración utilizando el sumatorio como estimación de dicha integral

```
static long num_steps=100000;
double step, pi;
void main() {
    int i;
    double x, sum=0.0;
    step=1.0/(double) num_steps;
    for (i=0;i<num_steps;i++){
        x=(i+0.5)*step;
        sum=sum+4.0/(1.0+x*x);
    }
    Pi=step*sum;
    printf("Pi=%f\n", pi);
}
```

2. El segundo, estadísticamente, mediante el método de Montecarlo, tomando aleatoriamente un gran número de puntos N y determinando si cada punto si cae dentro del círculo $x^2+y^2 < R^2$, con $R=1$.

A continuación, se muestra una posible implementación secuencial de este método:

```
int main (int argc, char *argv[]) {
    unsigned short xi[3] = { 1, 2, 3 };
    unsigned long long count = 0;
    unsigned long long i;
    unsigned long long samples;
    double x, y;
    samples=3000000; /*Valor por defecto*/

    if (argc >1)
        samples = atoll(argv[1]);

    for (i = 0; i < samples; ++i) {
        x = erand48(xi);
        y = erand48(xi);
        if (x*x+y*y <= 1.0) {
            ++count;
        }
    }

    printf ("Valor estimado de pi:  %.7f\n", 4.0*count/samples);
    return 0;
}
```

Metodología

Para la paralelización se utilizará:

- OpenMP
- MPI
- Todas las técnicas de paralelismo estudiada en la asignatura

Presentación de Resultados

Deberá realizarse en grupos de 2 personas, y consistirá en una breve defensa oral que durará aprox. 10 minutos más preguntas (a ambos miembros del grupo). Se usará una breve presentación Power Point o similar. Debe/n explicarse los códigos y resultados obtenidos, preferiblemente de una máquina diferente a la de los laboratorios. La puntuación obtenida será individualizada, según la presentación y las respuestas de cada miembro del grupo. Se valorará:

- El uso de una máquina diferente, cuyas características deben describirse en la presentación. Es necesario describir toda la configuración usada, tanto hardware (procesador, memoria) como software (sistema operativo, compilador, etc.). Los experimentos deben poder ser replicables.
- El número de códigos diferentes analizados y comparados. En particular, en el caso del método de Montecarlo, los generadores de números aleatorios usados.
- El uso de métricas cuantitativas para la comparación en términos de aceleración, IA, precisión en la estimación del valor de π , etc.
- El nivel de detalle de las explicaciones realizadas, incluyendo gráficas comparativas de tiempos de ejecución, uso de memoria en relación al número de trapecios o pasos, etc.
- La calidad de la presentación y de las respuestas a las preguntas del profesor/a.
- Es importante centrarse en los aspectos más importantes, no repetir los enunciados. Por ejemplo: cómo se ha paralelizado el código, qué variables deben ser private o shared, qué tiempos de ejecución y aceleraciones se obtienen, qué dificultades se han observado, etc.

Entrega

Se deberá entregar en un fichero ZIP, identificado con el grupo de prácticas y el número de grupo (Por ejemplo, **Trabajo_L1_G3.zip**) la presentación, carpeta con códigos implementados y hojas de cálculo con los resultados presentados.

Extras de carácter voluntario a valorar

- La combinación de OpenMP y MPI
- La ejecución en clúster.

Se realizará prueba en un clúster real de la parte MPI o/y de OpenMP y MPI.

Para ello, se debe comunicar a la profesora M^a José Morón Fernández mediante email (mjmoron@us.es), para poder asignarle usuario y contraseña de acceso.

Previamente a la ejecución en el cluster, es altamente recomendable tener depurados y probados los ejercicios que se vayan a ejecutar en el mismo.

Para la ejecución de MPI en varios hosts, se deben seguir los pasos indicados en el Anexo del presente guion.

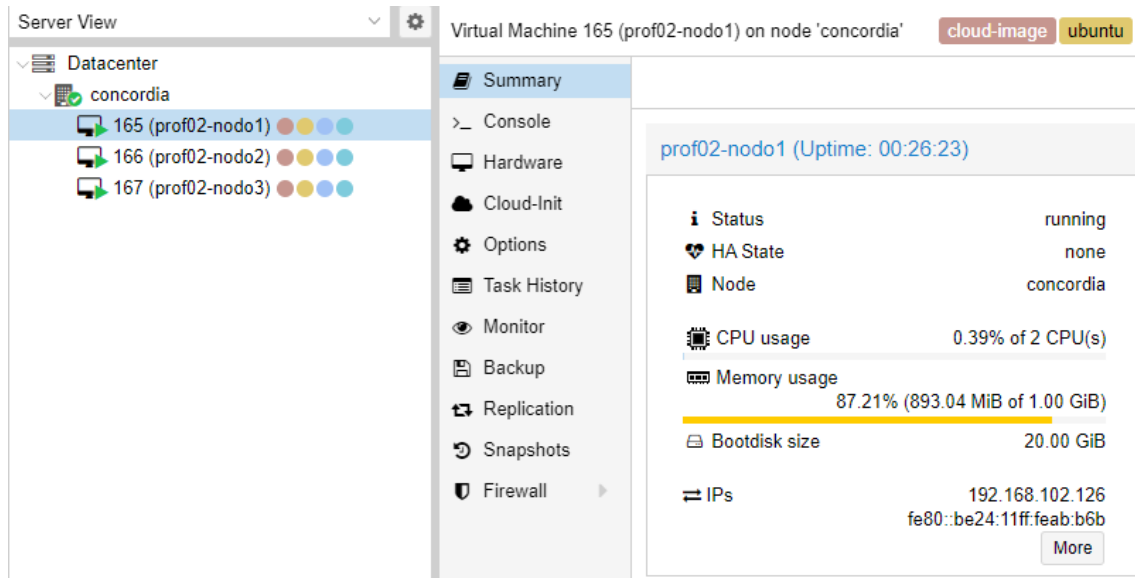
- La implementación mediante CUDA

Anexo I: Ejecución en el Clúster

Disponemos de tres MVs con S.O. Linux y, por ejemplo, las siguientes IPs:

- Nodo 1: 192.168.102.126
- Nodo 2: 192.168.102.127
- Nodo 2: 192.168.102.128

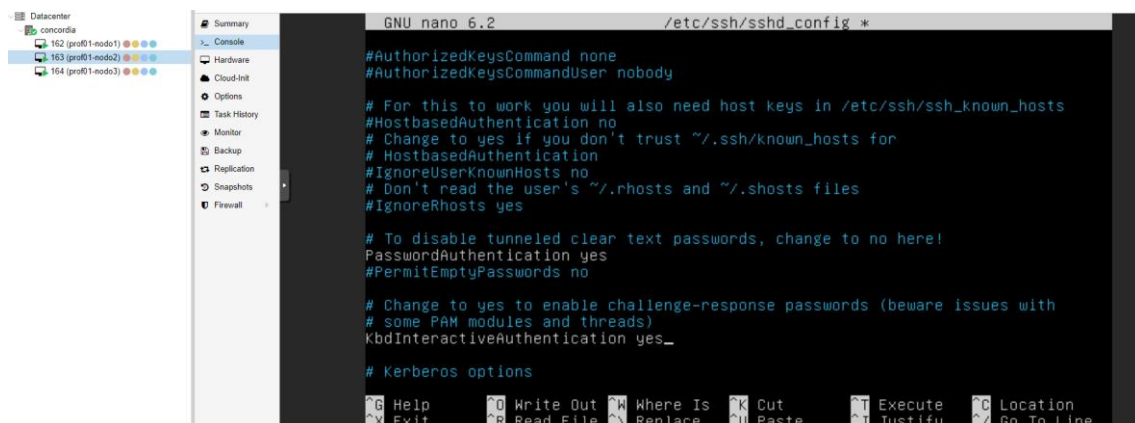
Se pueden consultar las características en la opción de “Summary”:



Los pasos a seguir para ejecutar un programa MPI en los tres nodos se detallan a continuación:

1. Modificar en cada uno de los nodos el fichero `/etc/ssh/sshd_config`:

```
sudo nano /etc/ssh/sshd_config:
```

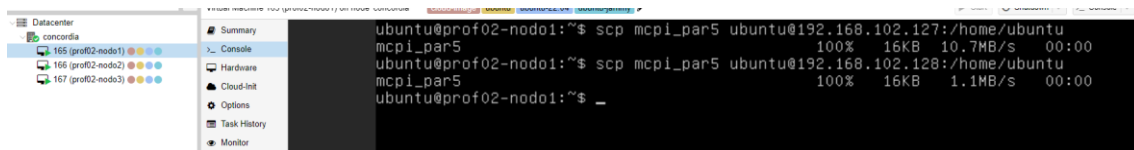


2. A continuación, reiniciar el servicio ssh:

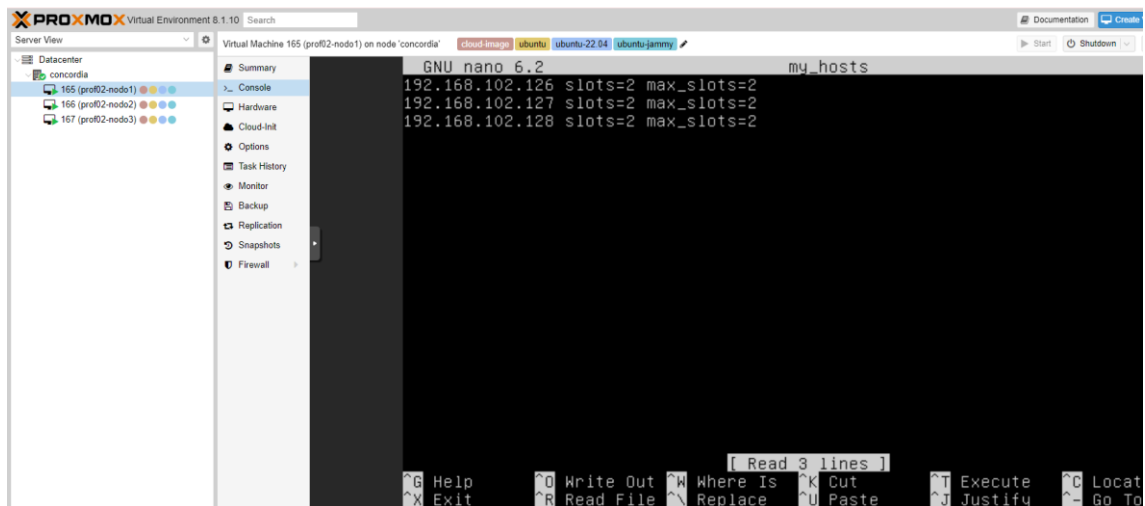
```
sudo service ssh restart
```

3. Copiamos el fichero a ejecutar en cada uno de los nodos mediante el comando scp:

```
scp mcpi_par5 ubuntu@192.168.102.127:/home/ubuntu
```



4. Creamos el fichero my_hosts: nano my_hosts



5. Ejecutamos mediante el comando mpirun:

```
mpirun -np 6 -hostfile my_hosts /home/ububtu/mcpi_par5
```