Versión MPI:

Ante la imposibilidad de convertir la escena completa a datos planos, cada hilo debe generar su propia escena. Con la función randomScene se ha comprobado que todos los hilos generan la misma escena, al tener la misma semilla.

Para solucionar esto se ha implementado una función que guarda la escena generada por el poroceso 0 en un txt, aprovechando la función implementada por el profesor para cargar dicha escena en los demás hilos.

Esta mejora no se refleja muy bien el tiempo de ejecución, aunque realmente si que existe una mejora en cuanto a la cantidad de cómputo realizado.

Puede que 10 procesos tarden 40 segundos en generar todos una imagen (proceso costoso) y tarden los mismos 40 segundos en esperar a que lo haga el proceso maestro y leer el archivo, pero lo segundo es mucho menos costoso en términos de cómputo.

En serie:

* Comp: g++ mainSerie.cpp Sphere.cpp Scene.cpp Metallic.cpp Crystalline.cpp utils.cpp random.cpp -o rayTracerSerie
* Ej: ./rayTracerSerie

Filas:

* Comp: mpic++ -O2 -std=c++17 -o raytracerMPIFil mainMPI\_Filas.cpp Crystalline.cpp Metallic.cpp random.cpp Scene.cpp Sphere.cpp utils.cpp -lm
* Ej: mpirun -np 8 ./raytracerMPIFil

Columnas:

* Comp: mpic++ -O2 -std=c++17 -o raytracerMPICol mainMPI\_Columnas.cpp Crystalline.cpp Metallic.cpp random.cpp Scene.cpp Sphere.cpp utils.cpp -lm
* Ej: mpirun -np 8 ./raytracerMPICol

Bloques:

* Comp: mpic++ -O2 -std=c++17 -o raytracerMPIBlocks mainMPI\_Bloques.cpp Crystalline.cpp Metallic.cpp random.cpp Scene.cpp Sphere.cpp utils.cpp -lm
* Ej: mpirun -np 8 ./raytracerMPIBlocks