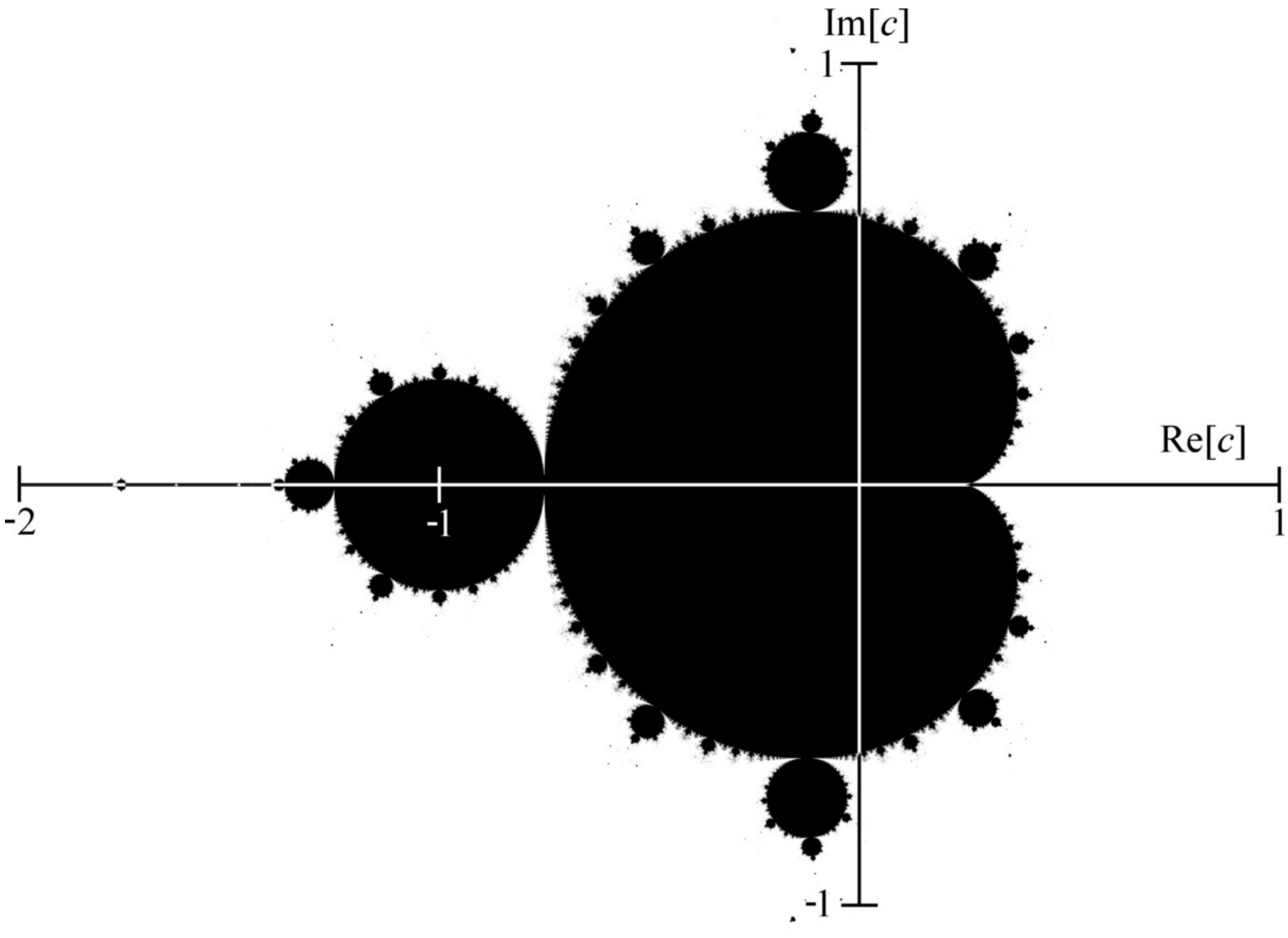
Cómputo Evolutivo – Segunda Parte – Tarea 1

Un fractal es un objeto geométrico cuya estructura básica se repite a diferentes escalas. Uno de los fractales más famosos es el conjunto de Mandelbrot, el cual está definido en el espacio de los números complejos. Específicamente, un número complejo c pertenece al conjunto de Mandelbrot, si al aplicar de forma recurrente la siguiente fórmula, el número no diverge, es decir, si su módulo no tiende a infinito:

$$z_{n+1} = z_n^2 + c$$

donde Z0 es igual a (0,0).

Hay que considerar que tanto Z como C son números complejos. Se puede demostrar que si durante la recursión se alcanza un número complejo cuya distancia al origen es superior a 2, dicha sucesión diverge, por lo que en ese momento ya se sabe que el número c con el que se comenzó no pertenece al conjunto de Mandelbrot. En la siguiente imagen, los puntos en negro son aquellos que pertenecen al conjunto de Mandelbrot, y se puede apreciar la forma fractal. En la misma el eje X es la parte real del número complejo y el eje Y es la parte imaginaria.



Generalmente, para determinar si un punto c pertenece al conjunto de Mandelbrot o no, se realiza mediante simulación. Dado el punto c, se aplica la recursión un número I de iteraciones. Si en algún momento se genera un número complejo cuya distancia al origen sea mayor que 2, se determina que dicho número no pertenece al conjunto de Mandelbrot, mientras que en caso contrario se determina que sí pertenece. Cuántas más iteraciones se deje, el conjunto será más parecido al conjunto real de Mandelbrot.

Para esta tarea, queremos crear imágenes de regiones del conjunto de Mandelbrot. Puede dibujar diferentes regiones y con diferentes números de píxeles, pero como base vamos a tomar la siguiente configuración:

- Para el eje X (parte real del número complejo), el rango va a ser: [-1.5, 0.5]
- Para el eje Y (parte imaginaria del número complejo), el rango va a ser: [-1.0, 1.0]
- La cantidad de puntos en ambos ejes van a ser: 1024
- El número de iteraciones se fija a 1,000,000

- Desarrolle un código secuencial para resolver el problema anterior. El código debe generar un archivo .pgm con el dibujo (ver aclaraciones). Los píxeles asociados a números complejos que sí pertenecen se pintarán en negro, mientras que los que no se pintarán en blanco.
- Desarrolle un código paralelo para resolver el problema anterior, utilizando el esquema maestro-trabajadores. En este esquema el maestro creará las tareas (calcular los números complejos que se deben evaluar) y se las enviará a los trabajadores. Piénsenlo para que el código sea lo más genérico posible, con el fin de poder reaprovechar su código si cambia la tarea.

Aclaraciones:
Teniendo en cuenta la forma en que se multiplican números complejos, la recurrencia que hay que calcular es:

$$Zr_{n+1} = Zr_n * Zr_n - Zi_n * Zi_n + Cr$$
$$Zi_{n+1} = 2 * Zr_n * Zi_n + Ci$$

donde Zr, Cr son las partes reales y Zi, Ci son las partes imaginarias.

Podemos calcular la distancia al origen con la siguiente fórmula:

$$d_n = \sqrt{(Zr_n^2 + Zi_n^2)}$$

La imagen .pgm se genera imprimiendo a un archivo lo siguiente:

P2
ancho altura
MaximoValor *//En este caso va a ser un 1*
Matriz de resultados *//Va a ser una matriz binaria*

Nota: la versión secuencial con los parámetros dados tarda algo más de 30 minutos, puede hacer pruebas preliminares con muy pocos píxeles, por ejemplo, 128 x 128.

Ejecuta la versión paralela utilizando 1 nodo C1 (12 procesos: 1 máster + 11 trabajadores).

Para la entrega suba el código secuencial y paralelo con comentarios, y un informe pequeño, indicando cómo organizó el códigos y los tiempos que obtuvo en el código secuencial y paralelo.
Evaluación:

- Código secuencial correcto: 2 puntos

- Código paralelo correcto: 5.5 puntos
- Informe: 1.5 puntos.
- Generalidad del código paralelo para ejecutar otros tipos de tareas: 1 punto

Fecha de entrega: 20 de octubre