ECCI-UCR
CI1323 Arq. de Computadoras
Tarea Programada Valor 8%

I Sem. 2016, L 14 de Marzo
Prof. Ileana Alpízar A
Entrega: J 31 Marzo, 11:59 pm

TRABAJO PARALELIZADO CON UNA MATRIZ DE ENTEROS v2

1. **Descripción del problema:**
   Debe diseñar y programar, un programa en lenguaje **C ó C++ con llamados a MPI**, de manera que se resuelva eficientemente (aprovechando de la mejor manera el paralelismo, y con una lógica eficiente), el siguiente trabajo con matrices:

   La idea es que el usuario defina las dimensiones de una matriz indicando en valor de **a** y de **c** , de manera que la matriz que se creará con valores enteros, aleatorios entre el **0 y el 4** inclusive, tenga **a x p filas** (en donde **p** es el número de procesos creados) y **c columnas**.
   El resultado final, que debe ser desplegado en pantalla por el **proceso 0**, será el siguiente:

   a. Número de veces que aparece cada uno de los 5 posibles valores enteros en cada una de las filas de la matriz.

   b. Número de veces que aparece cada uno de los 5 posibles valores enteros en cada una de las columnas.

   c. Sumatoria de los valores para cada columna de la matriz.

2. **Forma de realizar la tarea:**

   a. Se trabajara de manera individual o en parejas como máximo número de estudiantes.

   b. Dado que se tomará muy en cuenta la **manera en la que se decidió paralelizar la solución**, esto debe ser analizado antes de diseñar la solución, y explicarlo claramente en la documentación (defender su decisión)

   c. **La matriz debe ser creada realmente**, pero no necesariamente debe estar completa en la memoria de ningún proceso específico. Se debe elegir la mejor manera de almacenarle en memoria para permitir un trabajo eficiente.

      ***NOTA MUY IMPORTANTE:** Las dos ideas que se quisieron indicar acá son: Una, que **es permitido para esta tarea,** crear la matriz de manera distribuida y mantenerse almacenada de esa manera. Y la segunda idea, es que: **primero se debe crear la matriz con sus valores y luego se debe recorrer de manera eficiente para resolver los punto a, b, y c indicados arriba en el punto 1**. O sea, **no es válido** si cada vez que se genera un valor aleatorio para almacenarse como una entrada de la matriz, se aumenten los contadores correspondientes.

   d. En todo momento, durante sus pruebas y para la corrida final con los datos de prueba que se indican más adelante, **debe correr su programa utilizando las colas que maneja en el cluster el "SGE"**

3. **Lo que se debe entregar en archivos digitales: (se envía a *arqui.g1@gmail.com*, *arqui.g2@gmail.com* ó *arqui.g3@gmail.com*)**

   a. Descripción de la forma en la que se decidió paralelizar la solución (indicando qué realiza cada proceso) y de la solución en sí, es decir la **lógica completa del programa** (use un diagrama de bloques-o de actividades).

   b. El código fuente **con una completa y clara documentación interna**

   c. **Un resultado de su programa corrido con "SGE" para a = 5, p= 20 y c= 10000** (estos valores lo analizaremos en clase, dependiendo de qué tan cómoda quede la presentación del resultado final)

   d. Lista **de problemas no resueltos** al tiempo de entrega e ideas sobre su solución

4. **Puntaje**:

Documentación	**10%**
Diseño, lógica y programación	**90%**