

1ra. Práctica Calificada del curso de Programación Paralela

Alumno: Huari Gutierrez, Marco Antonio.

Código: 17200229.

Teoría

1) Explique con sus palabras ¿Qué es un proceso de una computadora?

Es una unidad de actividad que se caracteriza por la ejecución de una secuencia de instrucciones, un estado actual, y un conjunto de recursos del sistema asociado.

2) Explique a qué se refieren cuando hablamos de una comunicación punto a punto entre 2 procesos, proponer un ejemplo en código.

Es una comunicación que involucra solo a 2 procesos, habrá un emisor y un receptor generalmente. Un ejemplo para la comunicación entre 2 procesos sería los comandos:

```
MPI_Send (*msg_buf_p, cantidad, MPI_Datatype, destino, etiqueta, comunicador);
```

```
MPI_Recv (*msg_buf_p, cantidad, MPI_Datatype, fuente, etiqueta, comunicador, estado);
```

3) ¿Qué es una memoria RAM (principal), Caché y Virtual? E indicar ¿Cómo funcionan?

Memoria RAM: La memoria RAM es la memoria principal de un dispositivo donde se almacena programas y datos informativos.

La memoria RAM es conocida como memoria volátil lo cual quiere decir que los datos no se guardan de manera permanente, es por ello, que cuando deja de existir una fuente de energía en el dispositivo la información se pierde. Asimismo, la memoria RAM puede ser reescrita y leída constantemente.

Memoria Caché: Es una memoria auxiliar, de alta velocidad, que está hecha para realizar copias de algunos archivos o datos a los que el sistema necesitará acceder rápidamente. Estos datos podrían estar almacenados en la memoria principal del móvil u online

Memoria Virtual: Es el uso combinado de memoria RAM en su computadora y espacio temporero en el disco duro. Cuando la memoria RAM es baja, la memoria virtual mueve datos desde la memoria RAM a un espacio llamado archivo de paginación. El movimiento de datos desde y hacia los archivos de paginación crea espacio en la memoria RAM para completar su tarea.

Si su computadora está falta de la memoria RAM necesaria para ejecutar una operación o programa, Windows utiliza la memoria virtual para compensar.

4) ¿En qué consiste la programación en Memoria Distribuida y la programación en Memoria Compartida?

En la memoria distribuida cada procesador tiene su propia memoria local. Se utiliza para el paso de mensajes para intercambiar datos. En cambio, en la memoria compartida, cada procesador o proceso tiene un único espacio de memoria. Todos los procesadores tienen acceso a la memoria a través de una red de conexión.

5) Describa en 3 líneas como máximo e indicar los parámetros de los siguientes comandos del MPI.

- a) MPI_Send** (*msg_buf_p, cantidad, MPI_Datatype, destino, etiqueta, comunicador)

Es una función de envío de mensaje bloqueante de un proceso de origen a uno de destino. Al ser bloqueante significa que hasta que el mensaje no haya sido enviado no se continúa con la ejecución.

- b) MPI_Recv** (*msg_buf_p, cantidad, MPI_Datatype, fuente, etiqueta, comunicador, estado)

Es una rutina o función que sirve para recibir el mensaje desde otro proceso que ha utilizado el MPI_Send. Esta función bloquea el proceso hasta que se reciba un mensaje con las características especificadas.

- c) MPI_Reduce** (*input_data_p, *output_data_p, cantidad, MPI_Datatype, MPI_Operador, destino, comunicador)

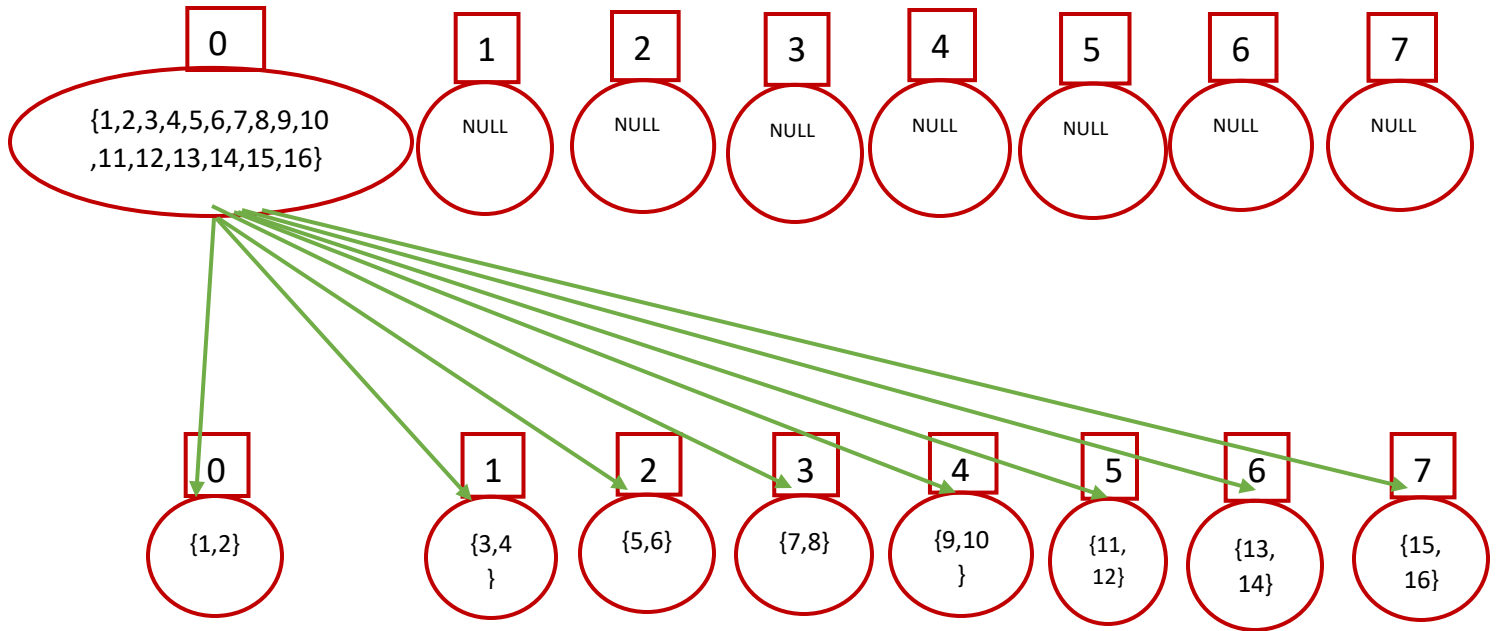
Realizan una operación de reducción global, por ejemplo: calcular máximo, mínimo, suma, AND lógico, etc. Sobre cada uno de los miembros del grupo. Después de realizar las operaciones entre un grupo de procesos, estos se reducen a un único proceso raíz.

- d) MPI_Allreduce** (*input_data_p, *output_data_p, cantidad, MPI_Datatype, MPI_Operador, comunicador)

Es igual al MPI_Reduce, a diferencia que el resultado se distribuye no solo a un único proceso raíz, sino que se distribuye a todos los procesos participantes.

Práctica

- 6) Utilizando MPI, implemente un algoritmo que determine el número de veces que un elemento x aparezca en un vector A con n elementos enteros. Se puede asumir que su algoritmo comienza con los elementos ya distribuidos entre los p procesos (n/p para cada uno).**
- 7) Desarrolle un algoritmo en MPI, utilizando p procesadores para calcular n!.**
- 8) Suponga que comm_sz=8 y la cantidad de elementos es n=16.**
- a) Diseñe un diagrama que explique cómo MPI_Scatter puede ser implementado usando comunicaciones basadas en árboles. Puede suponer que el origen del scatter es el proceso con rank 0.**



b) Hacer lo mismo para el MPI_Gather, en este caso con el proceso 0 como destino.

