

1ra Practica Calificada del curso de Programación Paralela

TEORIA

1. Explique con sus propias palabras ¿Que es un proceso en una computadora?

Un proceso, en informática, puede entenderse informalmente como un programa en ejecución. Formalmente un proceso 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 asociados".

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

Comunicación punto a punto

- Entre pares de procesos
- La selección de los mensajes es mediante ranking dentro de un grupo y tag (contexto).
- El ranking y tag están asociados a un comunicador

```
// Find out rank, size
int world_rank;
MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);
int world_size;
MPI_Comm_size(MPI_COMM_WORLD, &world_size);

int number;
if (world_rank == 0) {
    number = -1;
    MPI_Send(&number, 1, MPI_INT, 1, 0, MPI_COMM_WORLD);
} else if (world_rank == 1) {
    MPI_Recv(&number, 1, MPI_INT, 0, 0, MPI_COMM_WORLD,
            MPI_STATUS_IGNORE);
    printf("Process 1 received number %d from process 0\n",
           number);
}
```

3. ¿Que es una memoria RAM, Cache y Virtual? E indicar ¿Como funcionan?

RAM

La memoria de acceso aleatorio (Random Access Memory, RAM) se utiliza como memoria de trabajo de computadoras y otros dispositivos para el sistema operativo, los programas y la mayor parte del software.

En la RAM se cargan todas las instrucciones que ejecuta la unidad central de procesamiento (procesador) y otras unidades del computador, además de contener los datos que manipulan los distintos programas.

MEMORIA CACHE

En informática, una caché es un componente de hardware o software que guarda datos para que las solicitudes futuras de esos datos se puedan atender con mayor rapidez; los datos almacenados en una caché pueden ser el resultado de un cálculo anterior o el duplicado de datos almacenados en otro lugar, generalmente, da velocidad de acceso más rápido.

MEMORIA VIRTUAL

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.

4. ¿En que consiste la programación en Memoria Distribuida y la programación en Memoria Compartida?

Programación en Memoria Compartida Un único espacio de memoria para todos los procesos. Tienen acceso a la memoria a través de una red de conexión

Programación en Memoria Distribuida: Cada procesador tiene su propia memoria local. Se utiliza paso de mensajes para intercambiar datos.

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

a) MPI_Send: Realiza el envío de un mensaje de un proceso fuente a otro destino.

```
MPI_Send(&a //referencia al dato a enviar
        ,1 // tamaño del dato a enviar
        ,MPI_INT // Tipo de dato que envias
        ,rank // rank del proceso destino
        ,0 //etiqueta
        ,MPI_COMM_WORLD); //Comunicador
```

b) MPI_Recv: Rutina de recibimiento de un mensaje desde un proceso.

```
MPI_Recv(&carta // Referencia al dato donde se almacenara lo recibido
        ,1 // numero de elementos máximo a recibir
        ,MPI_INT // Tipo de dato que recibe
        ,rank // pid del proceso origen de la que se recibe
        ,MPI_ANY_TAG // etiqueta
        ,MPI_COMM_WORLD // Comunicador
        ,&estado); // estructura informativa del estado
```

c) MPI_Reduce: Reduce un valor de un grupo de procesos en un único proceso raíz.

```
MPI_Reduce(&in// Referencia al dato local
        ,&out //Referencia al dato donde se reducira el valor de entrada
        ,1 // numero de elementos máximo a recibir
        ,MPI_INT // Tipo de dato que recibe
        ,MPI_Op // Operacion a realizar con los datos de entrada
        ,rank // pid del proceso origen de la que se recibe
        ,MPI_COMM_WORLD );// Comunicador
```

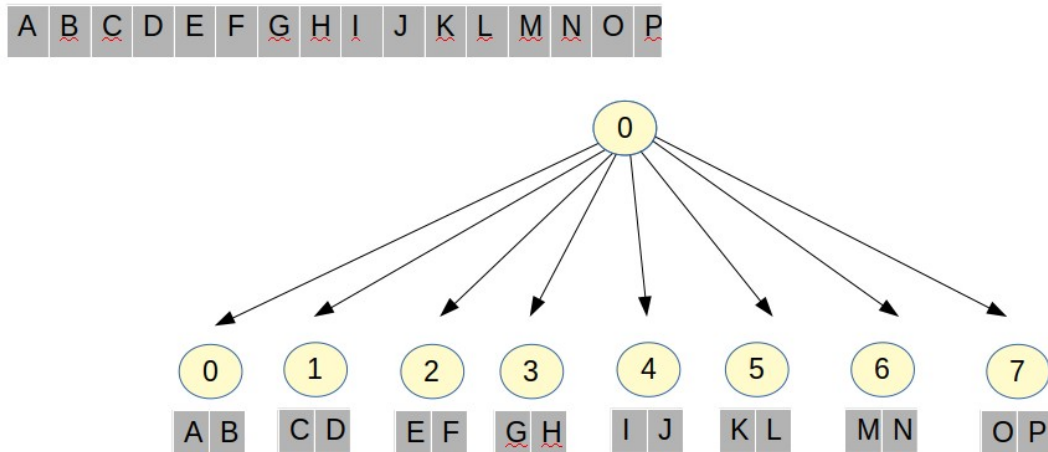
d)MPI_AllReduce: Reduce un valor de un grupo de procesos y lo redistribuye entre todos.

```
MPI_Reduce(&in// Referencia al dato local
        ,&out //Referencia al dato donde se reducira el valor de entrada
        ,1 // numero de elementos máximo a recibir
        ,MPI_INT // Tipo de dato que recibe
        ,MPI_Op // Operacion a realizar con los datos de entrada
        ,MPI_COMM_WORLD );// Comunicador
```

PRACTICA

5. Suponga que $\text{comm_sz}=8$ y la cantidad de elementos es $n=16$

a) Diseñe un diagrama que explique como MPI_Scatter puede ser implementado usando comunicaciones basadas en arboles. Puede suponer que el origen del scatter es el proceso con rank.



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

