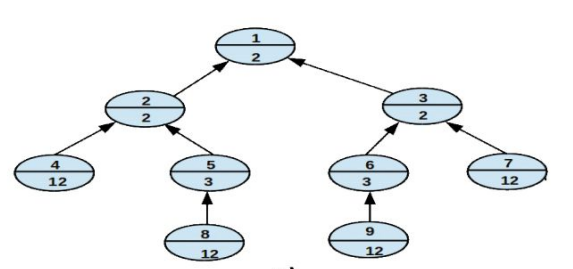
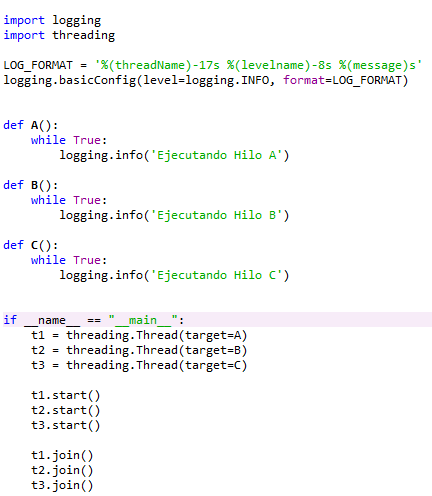
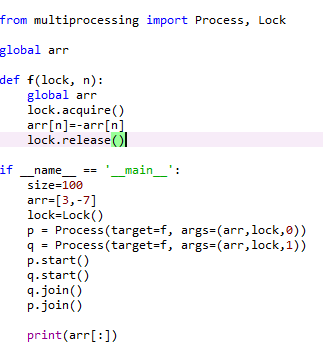
1. Define los siguientes conceptos:
   1. Descomposición y tareas
   2. Granularidad, granularidad fina y granularidad gruesa de una descomposición
2. Sea un sistema que posee un 25% del código secuencial y el 75% perfectamente paralelizable.
   1. Calcular el speedup para 8 procesadores y para 16 procesadores
   2. Si queremos ejecutarlo en 16 procesadores y queremos que la eficiencia sea del 80% (0.8) ¿Qué porcentaje de código debería ser paralelizable?
3. Una aplicación se ejecuta en una máquina paralela de 8 procesadores. Durante 50 segundos se utilizan los 8 procesadores, durante 20 segundos se utilizan 4, y durante 30 segundos sólo 1. La carga que se reparte a los procesadores está equilibrada, aunque en el caso de la ejecución en paralelo ha habido una sobrecarga (overhead) del orden del 10% del tiempo de ejecución. Haz una estimación del factor de aceleración (speed-up) que se consigue en relación con la ejecución serie y de la eficiencia obtenida.
4. Dado el siguiente grafo de dependencia, en el que cada nodo está etiquetado con su identificador en la parte superior y su coste en la parte inferior, calcula:



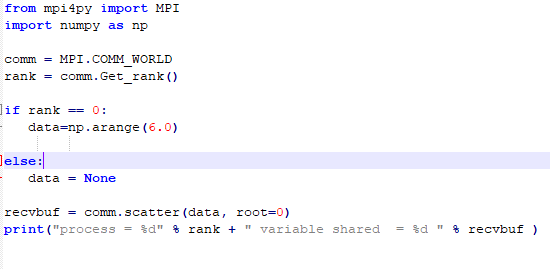
1. Máximo grado de concurrencia
2. Camino crítico
3. Grado medio de concurrencia
4. Máximo speedup posible y mínimo número de procesadores para conseguirlo. Indica la asignación de tareas a procesadores para conseguir dicho speedup.
5. Regula el siguiente código con semáforos de forma que la salida sea ABBCABBCABBC……. (sólo puedes definir semáforos y hacer llamadas a aquire y release en el código. El resto de modificaciones no están permitidas)



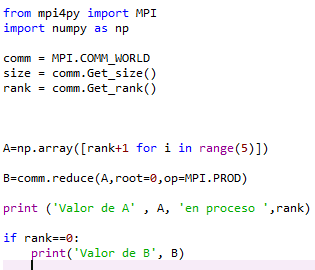
1. Escribe un pequeño algoritmo paralelo que calcule el doble de los números del 1 al 10 aprovechando al máximo los recursos de la máquina. Justifica las clases elegidas para la implementación y explícalas.
2. Dado el siguiente código:



1. Explica que hace el código y la salida que genera
2. ¿Están bien comunicados los dos procesos del código? Justifica tu respuesta realizando las modificaciones que consideres oportunas en el código.
3. ¿Están bien sincronizados los dos procesos del código? Justifica tu respuesta e indica si harías alguna modificación/optimización en el código.
4. Dado el siguiente código



1. Explica la salida que genera si se ejecuta con 4 y 6 procesadores.
2. Explica dos alternativas para que funcione correctamente con 3 procesadores.
3. Explica las diferencias entre las funciones de MPI scatter y Scatterv
4. Dado el siguiente código:



* 1. Indica la salida que genera al ejecutarlo con 3,6 y 9 procesadores. Explica que es la función de reducción en MPI y su funcionamiento.

1. Explica los siguientes conceptos de CUDA: kernel, host, device, grid, bloque e hilo.