

Computación de Altas Prestaciones

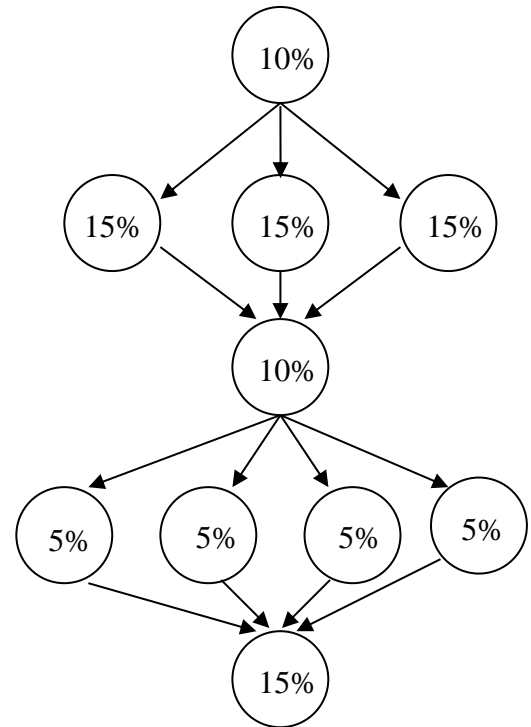
Computación Paralela

Problemas Tema 1: Paralelismo y evaluación de prestaciones

1.1.- En la figura se representa el grafo de dependencias entre las tareas de una aplicación. Para cada tarea se indica la fracción de tiempo de ejecución secuencial que tarda. Si se supone que el tiempo de ejecución secuencial en un procesador es de 60 segundos y que las tareas no pueden dividirse en subtareas de menor granularidad y no hay penalización por las comunicaciones. Se pide obtener el tiempo de ejecución en paralelo y la aceleración obtenida con:

- 4 procesadores.
- 2 procesadores.

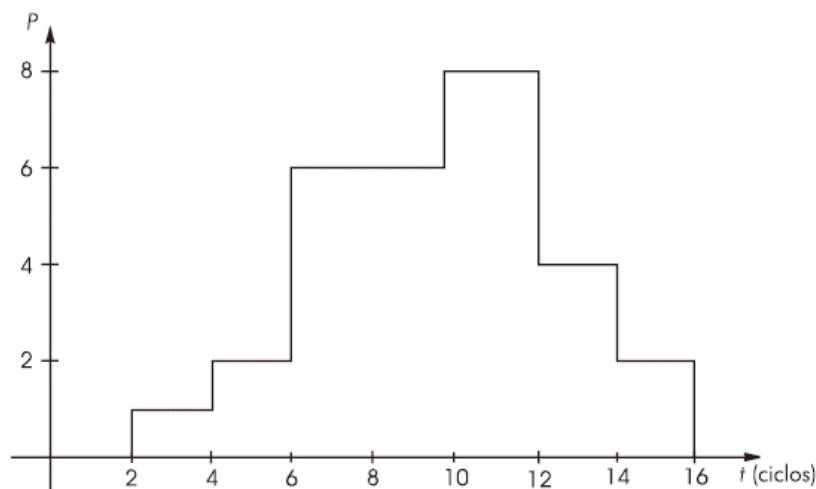
Represente el perfil de paralelismo del programa y el paralelismo real en función del tiempo que se consigue en cada caso.



1.2.- Un 25% de un programa no se puede paralelizar, el resto se puede distribuir por igual entre cualquier número de procesadores sin considerar sobrecarga.

- a) ¿Cuál es el máximo valor de aceleración que se podría conseguir al paralelizarlo?
- b) ¿A partir de qué número de procesadores se podría conseguir aceleraciones mayores o igual que 2?

1.3.- Supongamos que un programa tiene el perfil de paralelismo que se muestra en la figura:



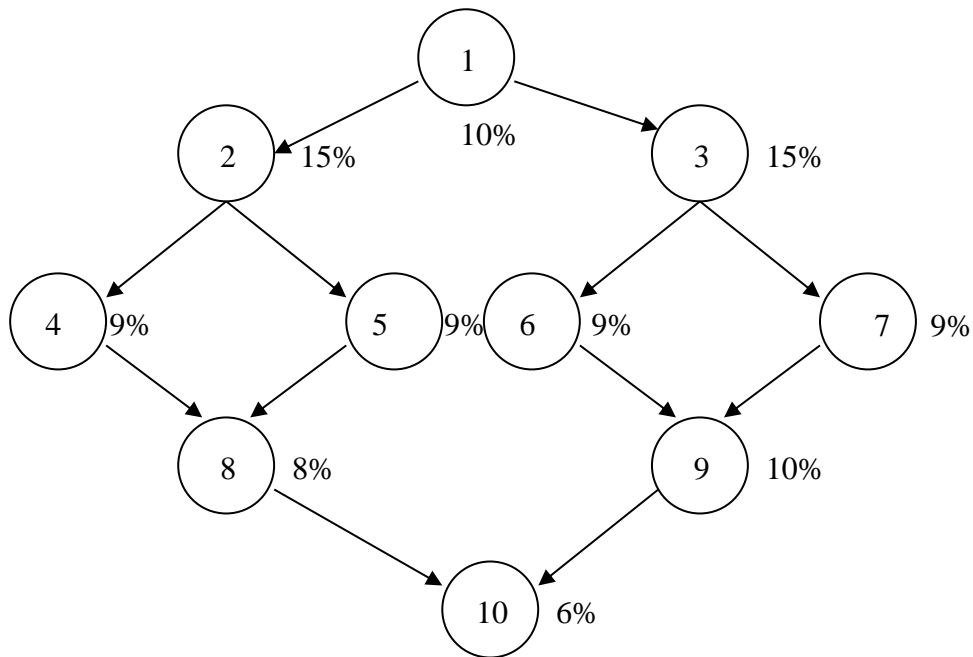
- a) Calcular el paralelismo medio del citado programa.
- b) Calcular la carga de trabajo de ese proceso si la gráfica de la figura se ha obtenido con procesadores que ejecutan 2 instrucciones por ciclo.

- c) ¿Cuál será la ganancia máxima de velocidad que podrá conseguirse cuando se ejecute ese programa en una máquina con 4 procesadores de la misma velocidad?
- d) En las condiciones de los apartados anteriores: ¿Cuál sería la ganancia máxima de velocidad que se obtendría, en ese sistema, con un programa que tuviera un grado de paralelismo constante de 8?

1.4.- Un programa, que tardaba 2 segundos en ejecutarse en un procesador, se ha dividido en 10 tareas. El orden de precedencia entre las tareas se representa en la siguiente figura, indicándose también el porcentaje de tiempo de cada una. Considerando que el tiempo de comunicación entre tareas se puede despreciar.

Represente el perfil de paralelismo del programa y calcule:

- b) ¿Qué tiempo tarda en ejecutarse el programa en paralelo?
- b) ¿Qué aceleración se obtiene con respecto a la ejecución secuencial?



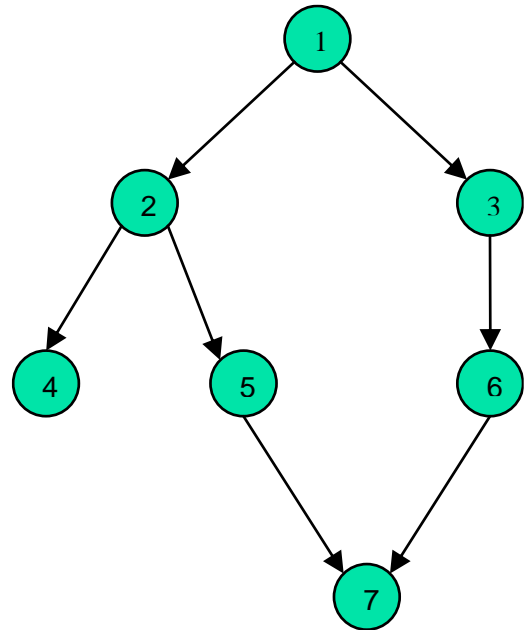
1.5.- La ejecución de un programa se distribuye en siete tareas con idéntico coste computacional y con las dependencias que se indican en el siguiente grafo.

En un sistema con 2 procesadores se planifican las tareas con el siguiente reparto;

P0	1	2	4	6	7
P1		3	5		

Calcule

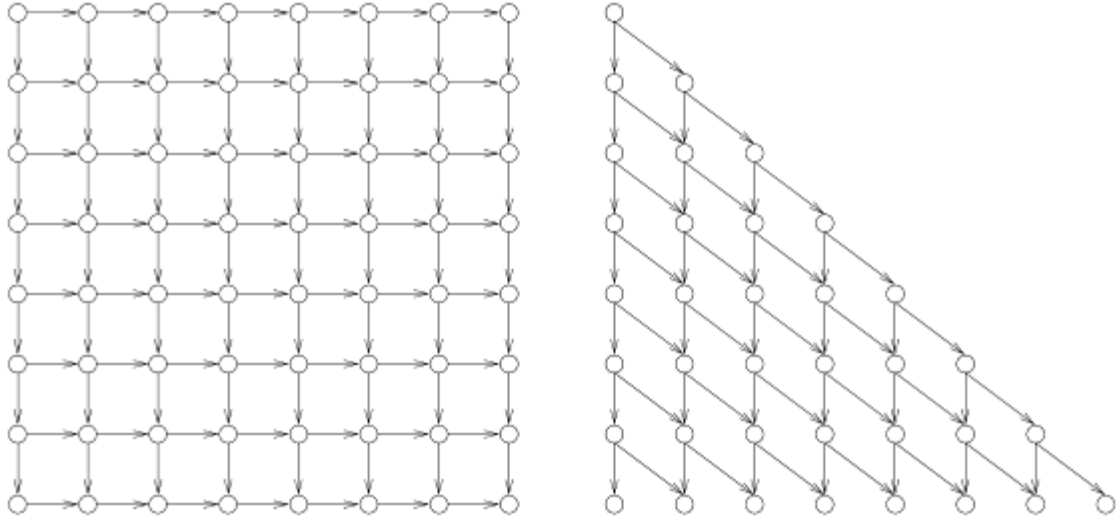
- Represente el perfil de paralelismo del programa
- Indique el tiempo de ejecución en un sistema con un solo procesador.
- La aceleración conseguida si se dispone de dos procesadores y se utiliza la planificación de la tabla.
- ¿Se puede obtener mayor aceleración para dos procesadores variando la planificación? ¿Qué criterio mejora la planificación?
- Cuantos procesadores son necesarios para alcanzar la máxima aceleración posible.



1.6.- La eficiencia de un sistema con n procesadores es de un 89%, resolviendo un problema que inicialmente se caracteriza por tener una 2% no paralelizable. Calcule el número de procesadores utilizados en el sistema bajo los siguientes supuestos.

- a) El trabajo computacional es de tamaño fijo y el sistema de n procesadores ha permitido reducir el tiempo de ejecución.
- b) El trabajo computacional se ha escalado con el número de procesadores para mejorar la precisión de la solución.

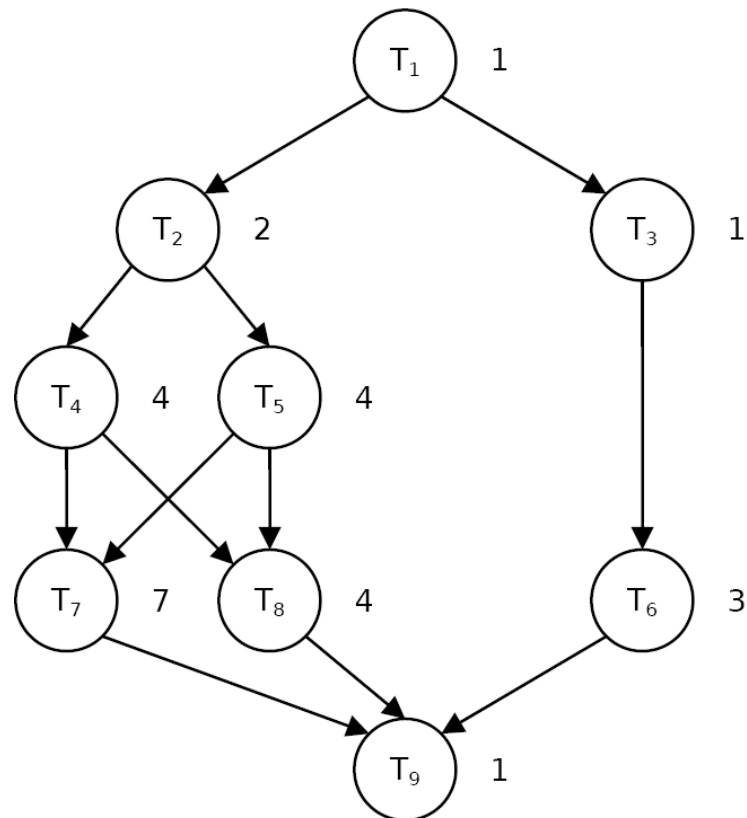
1.7.- La figura representan el grafo de dependencias de una aplicación que se va a ejecutar en un sistema multiprocesador con p procesadores. Si se generaliza denominando N al número de nodos en un grafo y n es un entero, de tal manera que en el grafo hay $N = n \times n$ nodos en la figura de la izquierda y $N = n \times (n+1)/2$ nodos en la de la derecha.



Se supone que no hay penalización por las comunicaciones y que cada nodo representa una tarea que tarda una unidad de tiempo y no puede dividirse en subtareas de menor granularidad.

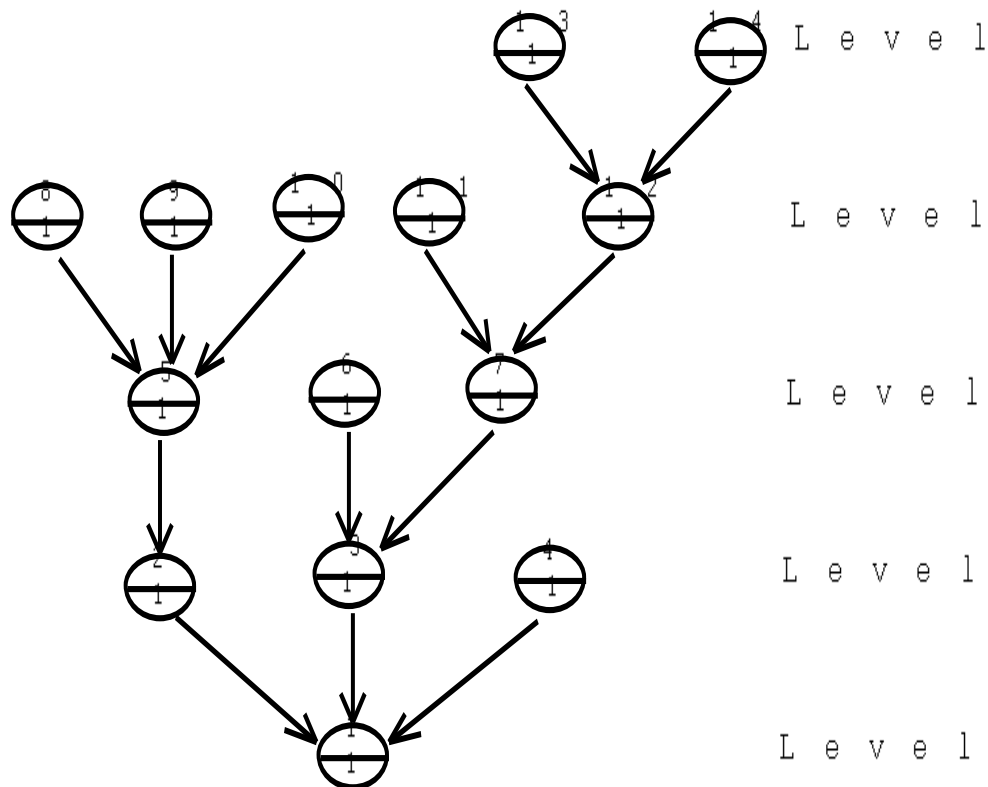
- Represente el perfil de paralelismo durante la ejecución de la aplicación para $n=8$
- El grado de paralelismo máximo y el trabajo realizado con grado de paralelismo tres.
- Calcule el speed-up (S) y la eficiencia (E) en función del número de procesadores $S(p)$ y compare para este caso los valores $S(\infty)$, $S(p=4)$ y $S(p=8)$
- Expresa el grado de paralelismo máximo y el Speed-up máximo en función de n .

1.8.- La figura representa el grafo de dependencias de una aplicación que se va a ejecutar en un sistema con p procesadores. Cada círculo representa una tarea independiente y el valor a su derecha representa el número de ciclos que tarda (i.e: T_5 tarda 4 ciclos, T_6 tarda 3 ciclos,...).



- e) Represente el perfil de paralelismo de esta aplicación.
- f) Calcule el grado de paralelismo máximo y el trabajo realizado con grado de paralelismo tres.
- g) Calcule el speed-up (S) o aceleración en función del número de procesadores $S(p)$ y compare para este caso los valores $S(\infty)$, $S(p=2)$ y $S(p=4)$.
- h) Calcule el número de procesadores para que la eficiencia del sistema sea máxima y en ese caso represente en un cronograma la distribución de tareas en función del tiempo para cada procesador.

1.9.- El grafo de dependencias de un algoritmo es el indicado en la figura. Cada tarea se ejecuta en un ciclo.



El esquema de planificación asigna a cada tarea una prioridad definida como el número de sucesores de cada nodo y a cada procesador disponible le asigna la tarea lista sin ejecutar con la más alta prioridad .

- Indique cual es el camino crítico que define la mayor longitud de planificación en unidades de tiempo.
- Estime el número de procesadores del sistema que optimiza la ejecución del algoritmo.
- Represente gráficamente el resultado de planificación en un sistema con 3 procesadores, indicando el tiempo final de ejecución.
- Repita el apartado anterior con 2 procesadores

1.10.- Considere el siguiente fragmento de código en C con sus comentarios y responda a los siguientes apartados:

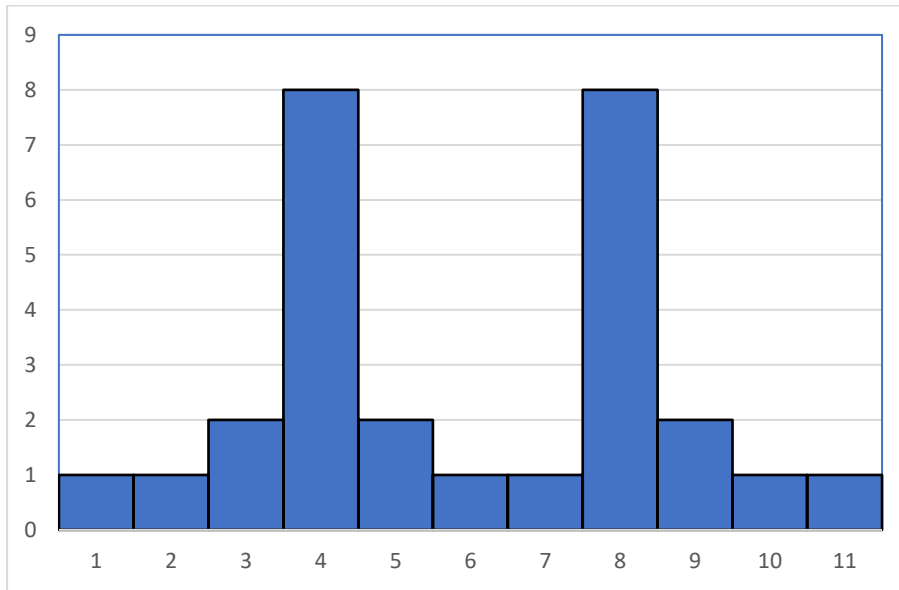
```
// Loop 1: it admits parallel implementation
#pragma omp parallel for
for(int i = 0; i < N; i++) {
    c[i] = 10 * a[i];
}

// Loop 2: it must be executed sequentially. We must wait for
// loop 1.
for(int i = 1; i < N; i++) {
    c[i] = 2 * c[i-1] * c[i];
}

// Final computation
result = c[0] * c[N-1];
```

- Dibuje el DAG de computación del fragmento de código con $N=8$, asumiendo que la operación elemental o tarea es el producto (operador binario $*$). Indique las tareas que no valgan la unidad si es que las hay.
- Dibuje el DAG de computación del fragmento de código en función de N , asumiendo que la operación elemental o tarea es el producto (operador binario $*$). Indique las tareas que no valgan la unidad si es que las hay.
- Según el grafo del apartado anterior, calcule el trabajo realizado, la profundidad máxima y el paralelismo medio en términos de N .
- Asumiendo infinitos procesadores, dibuje el perfil de paralelismo para $N=8$.
- Calcule el *speedup* y la eficiencia para $N=8$ con número de procesadores $p=2, 8$.
- Calcule el *speedup* con infinitos procesadores en función de N .

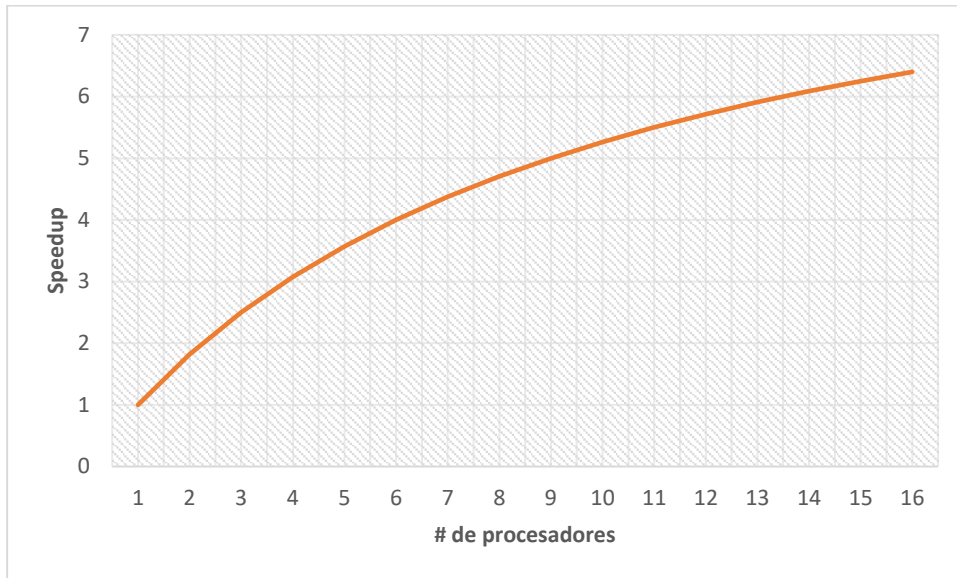
1.11.- Considere el siguiente perfil de paralelismo obtenido con 8 procesadores:



Considere que las tareas que se ejecutan en los **instantes 4 y 8**, son tareas que admiten escalado infinito, esto es, siempre **se ejecutarán con tantos procesadores como estén disponibles**.

- Indique qué modelo de paralelismo (WC, TC o MC) se ajusta más con este problema.
- Calcule el speedup para el caso $p=8$ procesadores.
- Calcule el trabajo realizado a $DOP=1$, $DOP=2$ y $DOP=p$.
- Calcule el speedup en función del número p de procesadores. ¿Se asemeja a alguna ley o teorema que conozca? Indique cuál e indique la relación.

1.12.- Se muestra a continuación el speedup para un programa concreto.



- ¿Qué ley o teorema está siendo representado?
- Asumiendo que el programa solo tiene dos tipos de trabajo: W_1 y W_n , es decir, trabajo con DOP=1 y con DOP=n, y que el trabajo total es $W = 100 \Delta$, calcule W_1 y W_n .
- ¿Cuáles son los requisitos para que la ley de Gustafson aplique? Dibujar el speedup que obtendría.