

Grados en Ingeniería Informática e Ingeniería de Computadores
Asignatura Algoritmos Avanzados
Curso 2022/2023
Práctica nº 1

Objetivo

El objetivo de la práctica es profundizar en la implementación de los algoritmos voraces.

Carácter

La realización de la práctica es voluntaria. Puede hacerse individualmente o en pareja.

Enunciado

Sea el siguiente problema de planificación de tareas. Se dispone de n procesadores del mismo modelo, $n > 0$, y de m tareas, $m > 0$. Cada procesador sólo puede ejecutar una tarea al mismo tiempo, necesitando cada tarea un tiempo t_i para ejecutarse. Todas las tareas se encuentran disponibles en el mismo instante. El tiempo de espera de cada tarea será el tiempo transcurrido mientras se ejecutan las tareas planificadas antes que ella en el mismo procesador más su propio tiempo de ejecución. Se pide desarrollar un algoritmo voraz que minimice el tiempo total de espera de todas las tareas.

Supongamos que las tareas están ordenadas en orden creciente de tiempo de ejecución. Un algoritmo voraz asigna la tarea 0 al procesador 0, la tarea 1 al procesador 1, y así sucesivamente de forma cíclica. Es decir, la tarea $n-1$ se asignará al procesador $n-1$, la tarea n al procesador 0, y así sucesivamente.

Por ejemplo, sean cinco tareas de duraciones $\{1, 2, 3, 4, 5\}$ y 2 procesadores. Se asignarán las tareas de duración 1, 3 y 5 al procesador 0 y las tareas de duración 2 y 4 al procesador 1.

El tiempo de espera de cada tarea será:

tarea de índice 0: 1 = 1
tarea de índice 1: 2 = 2
tarea de índice 2: 1+3 = 4
tarea de índice 3: 2+4 = 6
tarea de índice 4: 1+3+5 = 9

Por tanto, el tiempo total de espera de todas las tareas será: $1+2+4+6+9 = 22$

Se desea implementar dicho algoritmo voraz, primero suponiendo que las tareas están ordenadas y después sin esta suposición. La cabecera del método debe ser la siguiente (pudiendo variar solamente el identificador de método):

```
public static int tareas (int[] ts, int n)
```

donde:

- el vector ts contiene los tiempos de ejecución de las tareas,
- el entero n es el número de procesadores disponibles

y el método devuelve el tiempo total de espera de todas las tareas.

Informe

El alumno debe entregar un informe con la estructura que se detalla a continuación. El código de los algoritmos no debe enviarse en ficheros separados, sino integrarse en el texto del informe. Sólo se debe incluir el código de los métodos pedidos y, si fuera el caso, de los métodos auxiliares que utilicen.

1. **Algoritmo idealizado.** Codificar el algoritmo voraz descrito, suponiendo que las tareas están ordenadas en orden creciente de tiempo de ejecución.
2. **Algoritmo realista.** Modificar el algoritmo anterior de forma que funcione independientemente del orden inicial de las tareas. El algoritmo debe utilizar la técnica de ordenación de candidatos descrita en la asignatura.
3. **Análisis de complejidad.** Analizar la complejidad en tiempo y en memoria de ambas versiones del algoritmo.
4. **Conclusiones.** Se explican las conclusiones obtenidas tras realizar la práctica. Estas conclusiones pueden consistir en una valoración de la técnica voraz o cualquier comentario sobre la práctica. Por ejemplo, pueden describirse incidencias que han dificultado su realización, sus aspectos más atractivos o más difíciles, sugerencias, etc.

Entrega

El alumno debe entregar el informe por medio del apartado de Evaluación del aula virtual. Si se tienen dificultades, puede enviarse por el correo del aula virtual con el asunto “Práctica 1”. El plazo de entrega es el domingo 2 de octubre de 2022, incluido.

Evaluación

Se evaluará la calidad de los algoritmos desarrollados y los análisis de complejidad realizados, así como la claridad del informe.