



DISEÑO Y ANÁLISIS DE ALGORITMOS

ENUNCIADO DE LA PRÁCTICA 2

ALGORITMOS VORACES

Dpto. de Lenguajes y Sistemas Informáticos

Descripción del problema

Sea un conjunto A de n actividades $\{a_0, a_1, \dots, a_{n-1}\}$ que necesitan utilizar un recurso común, p.ej. una sala de reuniones. El recurso sólo puede ser usado por una actividad en cada momento. Cada actividad tiene un instante de comienzo c_i y un instante de finalización f_i , donde $0 \leq c_i < f_i < \infty$. Si se selecciona la actividad a_i , se desarrolla en el intervalo semiabierto de tiempo $[c_i, f_i)$. Las actividades a_i y a_j son compatibles si sus intervalos $[c_i, f_i)$ y $[c_j, f_j)$ no se solapan, es decir, si $c_i \geq f_j$ o $c_j \geq f_i$. El problema de selección de actividades consiste en determinar un subconjunto de actividades compatibles cuya cardinalidad sea máxima.

Por ejemplo, sea el siguiente conjunto de actividades:

i	0	1	2	3	4	5	6	7	8	9	10
c_i	1	2	0	5	8	5	6	8	3	2	12
f_i	4	13	6	7	12	9	10	11	8	5	14

Un subconjunto S de actividades compatibles es $\{a_2, a_4, a_{10}\}$. Sin embargo, no es un subconjunto de cardinalidad máxima, como lo son $\{a_0, a_3, a_7, a_{10}\}$ y $\{a_9, a_3, a_4, a_{10}\}$.

Se pide codificar un algoritmo voraz que encuentre la solución óptima al problema (si hay varias con una sola es suficiente).

La práctica se debe realizar en parejas y el lenguaje de programación es Java.

Descripción de la entrada

Los datos de la entrada pueden ser a través de un fichero en el que se indica en la primer línea el número de tareas y en las sucesivas líneas (una por cada tarea) se indica el rango de las tareas indicando el comienzo espacio en blanco y el final. Otra forma de recoger los datos de entrada es solicitándolo por teclado al usuario.

Descripción de la salida

La salida puede ser por pantalla o en un fichero y mostrará el índice de las tareas seleccionadas.

Ejecución del programa

El ejecutable se invocará con el siguiente formato:

planifica [fichero_entrada] [fichero_salida]

Los argumentos son los siguientes:

fichero_entrada: es el nombre del fichero del que se leen los datos de entrada. Si no existe el fichero programa dará error.

fichero_salida: es el nombre del fichero que se creará y almacenará la salida. Si el fichero ya existe, el comando dará un error. Si falta el fichero de salida, el programa muestra el resultado por pantalla.

El programa deberá poder recoger la entrada a través del teclado o de un fichero de entrada y podrá generar la salida en un fichero o por pantalla.

Tarea a desarrollar

Implementar un algoritmo para resolver el problema diseñado mediante la estrategia Voraz y realización de una memoria con los siguientes puntos:

- Portada (datos de identificación de la práctica y del alumno)
- Algoritmo propuesto
- Estudio teórico de la complejidad computacional del algoritmo
- Estudio empírico del comportamiento computacional del algoritmo. Estudiar cómo varía el comportamiento del algoritmo respecto al tamaño de entrada y compararlo con el estudio teórico. Se recomienda crear una gráfica del comportamiento del algoritmo para diferentes tamaños de problema.
- Alternativas al esquema utilizado: analizar otras opciones algorítmicas y si las hay comparar su coste con el de la solución realizada
- Descripción de los datos de prueba utilizados y de los resultados obtenidos con ellos

La práctica se realizará en parejas y ambos alumnos deberán conocer y responder a cualquier pregunta sobre la práctica presentada independientemente de quien haya desarrollado cada parte.

Entregables y fechas

Se entregará en un .zip o .rar (que se llamará con el nombre y apellidos de los dos integrantes de la pareja) los siguiente: el .jar, ficheros de prueba, los fuentes y la memoria en formato electrónico por el aula virtual. El nombre del fichero .zip o .rar debe seguir el siguiente formato: “NombreA_Apellido1ºA_NombreB_Apellido1ºB”, donde son respectivamente el nombre y “primer” apellido de los alumnos. Se realizará una presentación de la práctica consistente en la ejecución del programa con el profesor el día de entrega en el laboratorio de informática y los alumnos responderán a cualquier pregunta sobre el contenido de la misma.

La fecha de entrega y presentación es el día 26 de abril.

Observaciones y evaluación

En el examen de teoría pueden presentarse alguna/s pregunta/s sobre la práctica.

Esta práctica es obligatoria y su nota puntúa en la asignatura.