

ALGORITMOS Y ESTRUCTURAS DE DATOS III
Trabajo Práctico N°2

Primera entrega: 7-5-2007, hasta las 19:30 horas

Segunda entrega: 28-5-2007, hasta las 19:30 horas

Ver información general sobre los Trabajos Prácticos en la página de la materia en Internet.

El *Booble art* es un programa que muestra el mapa de Buenos Aires y sus atracciones turísticas. A medida que vamos recorriendo el mapa en la pantalla, podemos hacer un click en un punto y nos muestra varias fotos que corresponden a lugares que están cerca del punto que marcamos.

Nuestro objetivo es realizar un algoritmo que, dado un punto en el plano (la pantalla), decida cuáles son las fotos a mostrar.

Para ésto, vamos a considerar los siguientes criterios:

- Cada foto tiene una posición en el plano dada por sus coordenadas.
- La pantalla tiene un tamaño fijo de $s * t$
- Al clickear en un punto del plano, las fotos que deben mostrarse son TODAS aquellas fotos que contienen a ese punto (es decir, el punto está dentro del área dado por las coordenadas de la foto).
- Las fotos pueden tener diferentes tamaños, intersectarse, estar contenidas una dentro de otra, etc.
- Por supuesto, al listar las fotos a mostrar, pueden haber fotos que se intersecan, etc. Esto no nos preocupa, sólo queremos conocer la lista de fotos a mostrarse en otra pantalla, donde otro programa (que se hará en Algoritmos 4) acomodará estas fotos

Vamos a realizar un preprocesamiento del conjunto de fotos para luego, cada vez que se realiza un click, podamos decidir qué fotos deben mostrarse en forma eficiente.

1. (a) Dada un conjunto de fotos, construir un *árbol de intervalos* usando **árboles red black** (será explicado en clase) con las operación *Insertar*.
(b) Indicar el orden de las operación *insertar* No es obligatorio que figure en el informe la justificación del orden de *insertar* pero se preguntará sobre esto durante el coloquio.
(c) Calcular la complejidad TEMPORAL Y ESPACIAL de construir el árbol de intervalos.
(d) Diseñar un algoritmo que, dado un punto en el plano, decida el conjunto de fotos a mostrar. Para esto, implementar la operación *buscarintersección*.
(e) Calcular la complejidad TEMPORAL Y ESPACIAL de realizar una consulta (llamamos "consulta" a clickear sobre un punto del plano).
(f) Analizar el algoritmo midiendo el tiempo de ejecución tanto para el armado del árbol como para consultas. Es decir, para diferentes conjuntos de fotos (instancias), realizar varias consultas.
2. Ahora, pensemos la siguiente variante. Cada tanto, el programa Booble art se actualiza y adiciona nuevas fotos a su mapa. Explicar cómo cambia el algoritmo anterior, sus funciones, etc. Calcular la complejidad de la función *insertar* para este caso.

Entrada **Tp2Ej1.in** Este archivo contiene varias instancias de entrada. La primera línea contiene el número de instancias. Cada instancia es definida de la siguiente manera. La primera línea contiene el número de fotos n , las siguientes n líneas contiene las coordenadas de las fotos: $x_0 \ x_1 \ y_0 \ y_1$, donde $x_0 \leq x_1$ y $y_0 \leq y_1$.

Tp2Ej4.in Este archivo contiene consultas para hacer sobre todas las instancias del archivo anterior. La primera línea contiene la cantidad de consultas (k) y las siguientes k líneas contienen las coordenadas del punto del plano de la forma $x \ y$.

NOTA: NO ES OBLIGATORIO QUE LA ENTRADA SE INGRESE POR LINEA DE COMANDOS, PERO EL QUE QUIERE PUEDE HACERLO DE AMBAS MANERAS.

Salida **Tp2Ej4.out** Por cada instancia, cada línea contiene una foto que se debe mostrar identificada por sus cuatro coordenadas. La salida de cada instancias está separada por un 0.

Bibliografía: Fotocopiadora: "Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Capítulo 13.

Sugerencia: Para buscar información sobre interval-trees pueden consultar en <http://www.dgp.toronto.edu/people/JamesStewart/378notes/22intervals/>