Examen. 19 de junio de 2.006

Resolver cada pregunta en una hoja distinta.
No hay que entregar esta hoja con el examen.

1. (2,25 puntos) Suponer una estructura de relaciones de equivalencia mediante punteros al padre. Mostrar un ejemplo de operaciones donde se vea la diferencia entre usar la estrategia de balanceo de árboles y no usarla. Mostrar otro ejemplo que muestre la diferencia entre usar compresión de caminos o no. En ambos casos, mostrar la evolución de las estructuras para cada operación, explicarlo y razonar cuál es mejor.

2. (2,75 puntos) Un videoclub está desarrollando una aplicación que permita un rápido acceso a las películas de las que disponen (aproximadamente unas cien mil) y no se tiene claro qué tipo de estructura utilizar. El acceso se realiza la mayor parte de las veces por el título de la película y éstas rara vez se insertan como nuevas o se quitan del catálogo y, cuando esto ocurre, se realiza en un proceso nocturno que no influye en el rendimiento de la aplicación. Otra de las operaciones más frecuentes es el listado ordenado de todas las películas del sistema.

   Analizar las ventajas e inconvenientes de usar las siguientes estructuras de datos en este problema: lista ordenada, árbol binario de búsqueda, árbol AVL, árbol B, árbol trie, relaciones de equivalencia y tablas de dispersión.

3. (3 puntos) En un examen, todos los estudiantes se colocan en fila. El empollón se sienta siempre en la primera y todos los demás se copian del alumno colocado inmediatamente delante. Siempre habrá pérdida de información en la copia. En concreto, la matriz **Q[a, b]** indica el porcentaje de información que se transmite cuando **a** copia directamente de **b**, entre 0 y 1 (es decir, si **b** saca **x**, **a** sacaría **x*Q[a, b]**).

   El alumno **a** pretende buscar su estrategia de copia óptima, es decir, una distribución de los alumnos en la fila que le dé la mayor nota posible. Escribe un algoritmo para resolver el problema, calculando la nota obtenida (suponiendo que el primero saca un 10) y la lista de los alumnos que tiene sentados delante.

4. (2 puntos) Escribir una especificación formal, usando el método axiomático, del TAD **Grafo dirigido**, suponiendo los siguientes tipos y operaciones predefinidos:

   - **Natural**: cero, esCero, sucesor, suma, esIgual, resta.
   - **Conjunto**: vacío, esVacío, inserta, suprime, esMiembro, cardinalidad, unión, intersección, diferencia, etc.

   Los nodos del grafo serán de tipo genérico **T**. Sólo hay una operación para insertar aristas. Se supone que los nodos se insertan al insertar una arista que los contenga. De esta manera, las operaciones que queremos definir son las siguientes:

   - **crear**: devuelve un grafo sin nodos.
   - **insArista**: dado un par de nodos y un grafo, inserta la arista correspondiente.
   - **existeArista**: consulta si existe o no una arista en un grafo.
   - **existeNodo**: consulta si existe o no un nodo en un grafo.
   - **gradoEntrada**: devuelve el grado de entrada de un nodo del grafo.
   - **nodos**: dado un grafo, devuelve un conjunto con los nodos del grafo.
   - **raíces**: devuelve un conjunto con todos los nodos que no tienen predecesores en el grafo.

   Escribir las 4 partes de la especificación del TAD (nombre, conjuntos, sintaxis y semántica).

Nota: La pregunta 4 no deben hacerla los alumnos que tengan aprobada la práctica de especificaciones formales.

Examen. 19 de junio de 2.006

Resolver cada pregunta en una hoja distinta.
No hay que entregar esta hoja con el examen.

1. (3 puntos) Considerar las siguientes recurrencias:

   1. $t(n)= t(n/2)+1$ 2. $t(n)= 2t(n/2)+1$ 3. $t(n)= 2t(n/2)+n$ 4. $t(n)= 3t(n/2)+n$
   5. $t(n)= 4t(n/2)+n$ 6. $t(n)= 7t(n/2)+n^2$ 7. $t(n)= 8t(n/2)+n^2$

   Responder a una de los dos siguientes preguntas, sin dar por supuestas las fórmulas maestras vistas en clase:

   a) (100%) Deducir a partir de dichas recurrencias una fórmula maestra que las incluya a todas, resolverla de manera general y aplicarla de manera particular a cada una para obtener su orden exacto. Para cada una de las recurrencias, indicar un algoritmo conocido cuyo tiempo de ejecución se corresponda con ella.
   b) (70%) Resolver individualmente cada recurrencia y obtener su orden exacto. Para cada una de las recurrencias, indicar un algoritmo conocido cuyo tiempo de ejecución se corresponda con ella.

2. (2 puntos) Disponemos de un tablero de juego que consta de **n** casillas consecutivas (tipo juego de la oca). Cada casilla tiene asociado un valor entero (positivo o negativo). Un jugador parte de una casilla inicial (casilla uno) y avanza a través de las casillas según la tirada de un dado. Cada tirada del dado es un valor entre 1 y 6. El juego termina cuando el jugador sobrepasa la casilla **n**. La puntuación obtenida es igual a la suma de los valores asociados a las casillas en las que el jugador ha ido cayendo con las tiradas del dado. El propósito del juego es obtener la mayor puntuación posible.

   Diseñar un algoritmo voraz que encuentre una secuencia de tiradas con la mejor puntuación posible. Aplicar el algoritmo al siguiente ejemplo: **n**= 15, **T**= (1, 3, 5, -4, -3, 8, 2, -1, -8, -1, -7, -2, -6, -3, 2). El algoritmo diseñado, ¿garantiza la solución óptima?

3. (2,5 puntos) Resolver el problema del ejercicio 2 de forma óptima por programación dinámica. Dar la fórmula recursiva del problema, con sus casos base, e indicar las tablas que usa el algoritmo. Mostar la ejecución sobre el ejemplo del ejercicio 2. A partir de las tablas, indicar para el ejemplo cómo se obtiene la secuencia de movimientos óptima.

4. (2,5 puntos) Resolver el problema del ejercicio 2 de forma óptima por ramificación y poda. Se deberá utilizar el esquema visto en clase, que se puede dar por supuesto. Definir la forma de representar los nodos, la manera de calcular las cotas, la generación de los descendientes de un nodo y las demás funciones genéricas del esquema.

Nota: Los alumnos que tengan aprobada la práctica 4 tienen convalidado el ejercicio 2.