

Algoritmos y Estructuras de Datos.

Examen Final. [12 de Febrero de 2004]

Ej. 1.- [Primitivas (20 puntos)] Escribir las funciones del TAD DICCIONARIO listadas a continuación, implementado por tabla de dispersión cerrada con resolución lineal de colisiones ANULA(D), INSERTA(x,D), MIEMBRO(x,D) y SUPRIME(x,D). *Escribir todos los tipos, definiciones, funciones y procedimientos auxiliares necesarios.*

Ej. 2.- [Ejercicios de programación (total 80 puntos)]

- a) Escribir un procedimiento `procedure KRONECKER(LA, LB: lista; var L: lista);` que, dadas dos listas $LA = \{a_1, a_2, \dots, a_N\}$ y $LB = \{b_1, b_2, \dots, b_M\}$ retorna otra lista L con $N \times M$ valores de la siguiente forma:

$$L = \{a_1b_1, a_1b_2, \dots, a_1b_M, \\ a_2b_1, a_2b_2, \dots, a_2b_M, \\ \dots \\ a_Nb_1, a_Nb_2, \dots, a_Nb_M\} \quad (1)$$

Por ejemplo, si $LA = \{1, 3, 5, 0\}$ y $LB = \{1, 2, 3, 4\}$ entonces $KRONECKER(LA, LB, L)$ debe hacer que

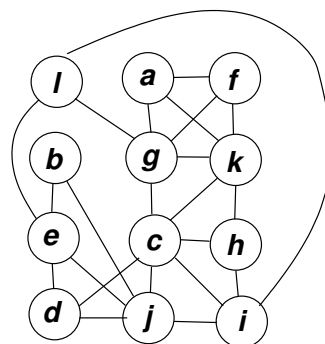
$$L = \{1, 2, 3, 4, 3, 6, 9, 12, 5, 10, 15, 20, 0, 0, 0, 0\}. \quad (2)$$

Usar las siguientes primitivas del TAD LISTA: INSERTA(x,p,L), RECUPERA(p,L), SUPRIME(p,L), SIGUIENTE(p,L), ANULA(L), PRIMERO(L), y FIN(L). No usar ninguna estructura auxiliar.

- b) **[cant-hojas (20 puntos)]** Escribir una función recursiva `function CANT_HOJAS(a:nodo; A:arbol):integer` que retorna la cantidad de hojas del árbol cuya raíz es a. Usar las funciones del TAD ARBOL ORDENADO ORIENTADO: HIJO_MAS_IZQ(n,A), HERMANO_DER(n,A), ETIQUETA(n,A).
- c) **[iguales (35 puntos)]** Escribir una función recursiva `function IGUALES(n,m: nodo; A: arbol) : boolean` que retorna verdadero (true) si dos árboles ordenados y orientados son iguales y falso (false) si no lo son. Dos rboles son iguales si tienen la misma estructura y contenido (etiquetas). Usar las funciones del TAD ARBOL ORDENADO ORIENTADO: HIJO_MAS_IZQ(n,A), HERMANO_DER(n,A), ETIQUETA(n,A).

Ej. 3.- [LIBRES] Ejercicios operativos (total 80 puntos)

- a) **[colorear-grafo (20 puntos)]** Colorear el siguiente grafo, utilizando una estrategia heurística para tratar de usar el menor número de colores posibles.



- b) **[reconstruir-arbol (20 puntos)]** Dibujar el árbol ordenado orientado cuyos nodos, listados en orden previo y posterior son
- $ORD_PRE = \{15, 22, 3, 14, 6, 7, 81, 9, 11, 12, 10, 5\}$.
 - $ORD_POST = \{22, 81, 7, 6, 12, 11, 9, 10, 14, 5, 3, 15\}$.
- c) **[abb (20 puntos)]** Dados los caracteres $\{z, t, a, l, r, k, b, h, w, x, n\}$ insertarlos, en ese orden, en un “árbol binario de búsqueda”. Mostrar las operaciones necesarias para eliminar los elementos z, k y l.
- d) **[tablas de dispersión (20 puntos)]** Insertar los números 9, 22, 32, 15, 14, 42, 24, 11, 34 en una tabla de dispersión cerrada con $B = 10$ cubetas, con función de dispersión $h(x) = x \bmod 10$ y estrategia de redispersión lineal.

Ej. 4.- [LIBRES] preguntas (total 20ptos, 5ptos/preg): [Responder según el sistema “multiple choice”, es decir marcar con una cruz el casillero apropiado. **Atención:** Algunas respuestas son intencionalmente “descabelladas” y tienen puntajes **negativos!!**]

- a) Las operaciones INSERTA y SUPRIME.MIN sobre el TAD COLA DE PRIORIDAD representado por montículos requiere un número de operaciones que está relacionado con ...
- ☐ ... la longitud del camino en el árbol desde la raíz hasta el punto de inserción o supresión.
 - ☐ ... el número de elementos en el montículo n
 - ☐ ... el cuadrado del número de elementos en el montículo n
 - ☐ ... el logaritmo de la semilla.
- b) El algoritmo de clasificación por urnas tiene un tiempo de ejecución $T(n) = O(n)$ pero está restringido a que el número de claves posibles ...
- ☐ ... sea finito.
 - ☐ ... sea delgado.
 - ☐ ... sea una potencia de 2.
 - ☐ ... sea infinito.
- c) La ventaja de la representación del TAD COLA por *arreglo circular* con respecto a la representación por celdas enlazadas es que ...
- ☐ ... requiere menos memoria ya que no necesita los punteros para enlazar celdas.
 - ☐ ... el orden de las operaciones FRENTE, QUITA, PONE es $O(\log n)$.
 - ☐ ... permite inserciones/supresiones en el medio de la cola.
 - ☐ ... es más simple.
- d) La búsqueda *exhaustiva* del camino de longitud mínima para el “Problema del Agente Viajante” (PAV) tiene un tiempo de ejecución... (n es el número de ciudades)
- ☐ $O(n)$
 - ☐ $O(1)$
 - ☐ $O(n^2)$