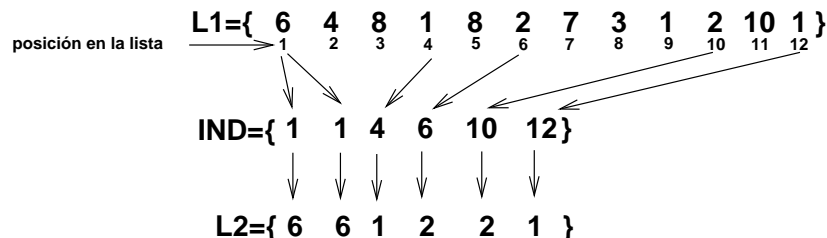


Algoritmos y Estructuras de Datos. Examen Final. [8 de Mayo de 2003]

Ej. 1.- Escribir las funciones funciones del TAD ARBOL ORDENADO ORIENTADO listadas a continuación, con celdas enlazadas por punteros o cursores:

- a) PADRE(n, A)
- b) HIJO_MAS_IZQ(n, A)
- c) HERMANO_DER(n, A)
- d) ETIQUETA(n, A)
- e) CREA2($v, A1, A2$)
- f) ANULA(A)

Ej. 2.- Escribir un procedimiento `procedure MULTI_REC(L1, IND: lista; var L2: lista);` que, dados una lista L1 y otra lista IND de índices enteros, retorna los elementos de la lista L1 que están en las posiciones indicadas por los números en IND. Se asume que los índices en IND son crecientes, es decir $IND_j \leq IND_{j+1}$, para todo j . Por ejemplo,



Usar las funciones del **TAD LISTA**: INSERTA(x, p, L), RECUPERA(p, L), SUPRIME(p, L), SIGUIENTE(p, L), ANULA(L), PRIMERO(L), y FIN(L).

Ej. 3.- Ejercicios básicos sobre TAD's

- a) Escribir una función "ACOTADO(n : nodo; A : arbol; max : integer): boolean;" que retorna true si todas las etiquetas del árbol binario A están por debajo de max y false en caso contrario. Usar las funciones del **TAD ARBOL BINARIO**: HIJO_IZQ(n, A), HIJO_DER(n, A), ETIQUETA(n, A).
- b) Escribir un procedimiento `procedure SEPARA(C: cola; var CPAR, CIMPARG: cola);` que separa los elementos de la cola C poniendo los pares en CPAR y los impares en CIMPARG. Utilizar las funciones del **TAD COLA**: ANULA(C), PONE_EN_COLA(x, C), QUITA_DE_COLA(C), VACIA(C), y FRENTE_DE_COLA(C).

Ej. 4.- [LIBRES] Ejercicios operativos:

- a) **Árboles:** Dibujar el árbol ordenado orientado cuyos nodos, listados en orden previo y posterior son
 - $ORD_PRE = \{C, Z, Q, U, V, W, X, R, T\}$.

Apellido y Nombre: _____

Carrera: _____ DNI: _____

[Llenar con letra mayúscula de imprenta GRANDE]

■ $\text{ORD_POST} = \{Z, U, X, W, C, Q, T, R, C\}$.

b) [LIBRES] Dados los enteros $\{1, 12, 8, 5, 7, 2, 3\}$ ordenarlos por el métodos de “montículos”. Mostrar el montículo antes y después de cada inserción.

c) [LIBRES] Preguntas: [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!!**]

1) Dadas las funciones

- $T_1(n) = 5\sqrt{n} + 2n^2$,
- $T_2(n) = n^3 + \log n$,
- $T_3(n) = \log n + 0.5 \cdot 2^n$ y
- $T_4(n) = 0.3 \log n + 7$

decir cuál de los siguientes ordenamientos es el correcto

- ☐ $T_2 < T_1 < T_4 < T_3$
- ☐ $T_4 < T_1 < T_2 < T_3$
- ☐ $T_2 < T_3 < T_4 < T_1$
- ☐ $T_1 < T_3 < T_2 < T_4$

2) ¿Cuál es la ventaja de las listas doblemente enlazadas con respecto a las simplemente enlazadas?

- ☐ $\text{ANTERIOR}(p, L)$ es $O(n^2)$.
- ☐ $\text{SIGUIENTE}(p, L)$ es $O(n)$.
- ☐ $\text{ANTERIOR}(p, L)$ es $O(1)$.
- ☐ $\text{LOCALIZA}(x, L)$ es $O(1)$.

3) ¿Cuál es el tiempo de ejecución del algoritmo de clasificación por montículos en el peor caso?

- ☐ $O(n \log n)$.
- ☐ $O(n^2)$.
- ☐ $O(1)$.
- ☐ $O(n^3)$.

4) El algoritmo de Huffman permite obtener códigos binarios para encodar mensajes utilizando árboles binarios. La longitud del código asignado a un caracter, en bits está dado por...

- ☐ ... la profundidad del nodo correspondiente en el árbol.
- ☐ ... la altura del nodo correspondiente en el árbol.
- ☐ ... la etiqueta del nodo correspondiente en el árbol.
- ☐ ... el número de nodos en el subárbol que cuelga del nodo correspondiente.