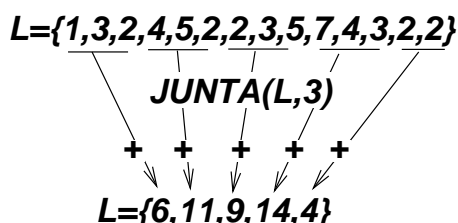


Algoritmos y Estructuras de Datos. Examen Final. [6 de Mayo de 2004]

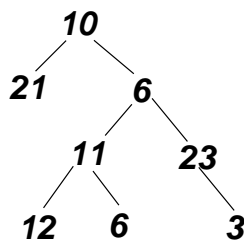
Ej. 1.- [primi (20 puntos)] Escribir las funciones del TAD LISTA con celdas simplemente enlazadas por punteros ó cursores. INSERTA(x, p, L), LOCALIZA(x, L), RECUPERA(p, L), SUPRIME(p, L), SIGUIENTE(p, L), ANULA(L), PRIMERO(L), y FIN(L). Escribir todos los tipos, definiciones, funciones y procedimientos auxiliares necesarios.

Ej. 2.- [Programacion (total 80 puntos)]

- a) **[junta (40 puntos)]** Escribir un procedimiento `procedure JUNTA(var L: lista; n: integer);` que dada una lista L , agrupa de a m elementos dejando su suma (ver figura). 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. Prestar atención a no usar posiciones inválidas después de una supresión. El algoritmo debe tener un tiempo de ejecución $O(n)$, donde n es el número de elementos en la lista original.



- b) **[maximo-par (20 puntos)]** Escribir una función `function MAXIMO_PAR(n: nodo; A: arbol): integer;` que retorna el máximo de las etiquetas **pares** de un árbol binario. En el caso del árbol de la figura debe retornar 12. Usar las primitivas del TAD ARBOL BINARIO: HIJO_IZQUIERDO(n, A), HIJO_DERECHO(n, A) y ETIQUETA(n, A),



- c) **[elimina-valor (20 puntos)]** Escribir un procedimiento `procedure ELIMINA_VALOR(var C: cola; n: integer);` que elimina todas las ocurrencias del valor n en la cola C . Por ejemplo, si $C = \{1, 3, 5, 4, 2, 3, 7, 3, 5\}$, después de `ELIMINA_VALOR(C, 3)` debe quedar $C = \{1, 5, 4, 2, 7, 5\}$. Sugerencia: Usar una estructura auxiliar lista o cola. Utilizar las primitivas del TAD COLA: ANULA(C), PONE_EN_COLA(x, C), QUITA_DE_COLA(C), VACIA(C), y FRENTE_DE_COLA(C). El algoritmo debe tener un tiempo de ejecución $O(n)$, donde n es el número de elementos en la cola original.

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

- a) **[arboles (30 puntos)]** Dibujar el árbol ordenado orientado cuyos nodos, listados en orden previo y posterior son

- $ORD_PRE = \{Q, T, R, S, V, W, Y, M, N\}$,
- $ORD_POST = \{T, S, M, N, Y, W, V, R, Q\}$.

b) **[huffman (30 puntos)]**: Dados los caracteres siguientes con sus correspondientes probabilidades, contruir el código binario y encodar la palabra TRANSVERSAL $P(T) = 0.4, P(R) = 0.2, P(A) = 0.1, P(N) = 0.1, P(S) = 0.05, P(V) = 0.05, P(E) = 0.05, P(L) = 0.05$ Calcular la longitud promedio del código obtenido.

Ej. 4.- [abb (20 pto)] Dados los enteros $\{4, 8, 11, 6, 5, 13, 23\}$ insertarlos, en ese orden, en un TAD ARBOL BINARIO DE BUSQUEDA. Mostrar las operaciones necesarias para eliminar los elementos 8, 11 y 6, en ese orden.

Ej. 5.- [LIBRES] [preguntas (total 20ptos, 5 pto c/una)] [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) Sea un tabla de dispersión abierta con B cubetas y n elementos. Asumiendo que la función de dispersión es lo suficientemente buena como para distribuir los elementos en forma uniforme entre las cubetas, el costo medio de inserción de un nuevo elemento es

- $O(n^2/B)$
- $O((n/B)^2)$
- $O(n + B)$
- $O(n/B)$

b) Dadas las funciones

- $T_1(n) = 3n^3 + 7n!$,
- $T_2(n) = 2n^2 + 0.5\sqrt{n}$,
- $T_3(n) = 7n^3 + 8!$ y
- $T_4(n) = 4^3 + 7^n$

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

decir cuál de los siguientes ordenamientos es el correcto

c) Sea una lista $L=(1,3,5,7)$ simplemente enlazada por punteros o cursores. Despues de hacer

```
p = PRIMERO(L);
p = SIGUIENTE(p,L);
p = SIGUIENTE(p,L);
SUPRIME(p,L);
p = SIGUIENTE(p,L);
```

¿Que retorna RECUPERA(p,L) ?

- ☐ ... retorna 1
- ☐ ... produce un error
- ☐ ... retorna 7
- ☐ ... retorna 3

d) ¿Cuál es el criterio para elegir una buena función de dispersión? ...

- ☐ Debe tratar de concentrar los elementos en pocas cubetas.
- ☐ Debe tratar de concentrar los elementos en una sólo cubeta.
- ☐ Debe tratar de concentrar los elementos en la primera cubeta.
- ☐ Debe distribuir los elementos en la forma más uniforme posible entre las cubetas.