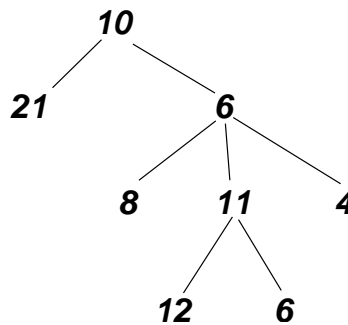


## Algoritmos y Estructuras de Datos. Examen Final. [9 de mayo de 2002]

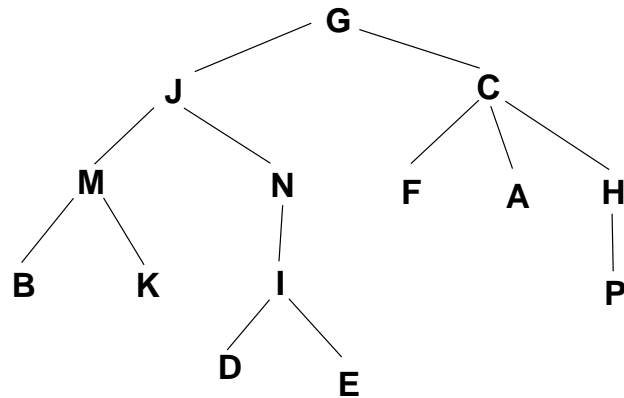
- Ej. 1.-** Escriba una implementación para el TAD CONJUNTO con listas simplemente enlazadas clasificadas implementadas con punteros o cursores. Las operaciones a implementar son: ANULA, INTERSECCION, UNION, DIFERENCIA, MIEMBRO, INSERTA y SUPRIME. Incluir los tipos y definiciones necesarios.
- Ej. 2.-** Escribir un procedimiento `procedure INTERCAMBIA(var L:lista);` que intercambia los elementos de L de tal forma que aquellos que están en una posición impar son intercambiados con el elemento siguiente. Es decir el primero es intercambiado con el segundo, el tercero con el cuarto y así siguiendo. Si el número de elementos es impar, entonces el último elemento queda inalterado. Así por ejemplo, si  $L = \{10, 1, 15, 7, 2, 19, 15, 16, 1\}$  entonces después de intercambiar debemos tener  $L = \{1, 10, 7, 15, 19, 2, 16, 15, 1\}$ .
- Utilizar las primitivas 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.-** Escribir una función `function SUMA_PAR_ANTEC(n:nodo;a:arbol) : integer;` que devuelve la suma de las etiquetas de los nodos tales que su etiqueta y la etiqueta de todos los antecesores es par. Se sugiere hacerlo en forma recursiva, notando que si la etiqueta de un nodo es impar, entonces `SUMA_PAR_ANTEC` de todo el subárbol correspondiente es nulo. Por, ejemplo para el árbol de la figura, `SUMA_PAR_ANTEC` da 28. Notar que las etiquetas 6 y 12 no son incluidas ya que su antecesor 11 es impar.



Utilizar las primitivas del **TAD ARBOL ORDENADO**: `PADRE(n,A)`, `HIJO_MAS_IZQ(n,A)`, `HERMANO_DER(n,A)`, `ETIQUETA(n,A)`, `RAIZ(n,A)`, `ANULA(L)`.

- Ej. 4.-** [LIBRES] Dados los enteros  $\{15, 3, 10, 9, 7, 4, 2\}$  ordenarlos por el método de "montículos". Mostrar el montículo antes y después de cada inserción.
- Ej. 5.-** [LIBRES] Dados los enteros  $\{10, 9, 12, 6, 14, 11, 7, 5, 13, 19, 2\}$  insertarlos, en ese orden, en un "árbol binario de búsqueda". Mostrar las operaciones necesarias para eliminar los elementos 10, 7 y 11.

Ej. 6.- [LIBRES] Dado el siguiente árbol



- (a) Listar en orden previo, posterior y simétrico las etiquetas.
- (b) ¿ Cuáles son las profundidades y alturas de los nodos G, N, E y C?