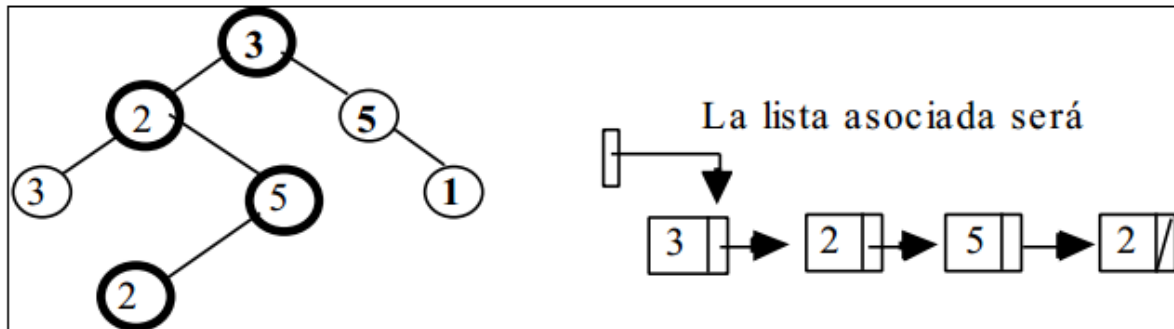


Guía Repaso Final

1. Diseñar y escribir un algoritmo tal que dado un árbol binario, devuelva una lista que contenga los elementos de la rama más larga del árbol de entrada. Si hubiera varias ramas con la misma profundidad, la lista contendría los elementos de una cualquiera de ellas.

Ejemplo:



2. Escribir un algoritmo que dada una lista de números decimales ingresados por teclado, se obtenga otra lista con los números en binario.
Este algoritmo consta de tres partes:
 - A. Creación de la lista con los números decimales ingresados por teclado.
 - B. Conversión de Decimal a Binario (uso de una función recursiva).
 - C. Mostrar por pantalla la lista con los números en binario.
3. Partiendo del siguiente array, encuentre las particiones e intercambios que realiza el algoritmo de ordenación quicksort para su ordenación.
8 → 43 → 17 → 6 → 40 → 16 → 18 → 97 → 11 → 7
4. Cree un método a la estructura árbol que imprima el nivel de la hoja que tiene el número más grande y el nivel de la hoja que tiene el número más chico.
5. Se tiene una lista en la que se han introducido pedidos según el orden de llegada, por lo que puede haber más de un pedido del mismo artículo. Se pide escribir un método de la estructura lista que devuelva un único elemento por cada artículo. El campo Cant_Pedida, miembro de la clase nodo, tiene la suma de todas las cantidades pedidas de un artículo determinado.
6. Escribir la versión iterativa de la Búsqueda Binaria. La misma recibe un arreglo arrayA[], el tamaño y el dato a buscar. Devuelve 1 si encontró el valor pedido y 0 si no lo encontró.
7. Dado una oración ingresada por teclado se pide realizar una función recursiva que reemplace en todo el texto las letras "a" por "e"

Ejemplo:

Oración original:

Los alumnos deben estudiar durante las vacaciones si desean rendir todas las materias.

Oración con reemplazo:

Los elumnos deben estudier durente las vececiones si deseen rendir todes las meteries.

8. Hacer una función que dada una frase ingresada por teclado la imprima invertida.
Implementar la función con la estructura de datos vista más adecuada.
Por Ejemplo: "una imagen vale por mil palabras" debe convertirse en "palabras mil por vale imagen una".
9. El recorrido en preorden de un determinado árbol binario es: GEAIBMCLDFKJH y en inorden IABEGLDCFMKHJ .
10.
Resolver:
 - a) Dibujar el árbol binario.
 - b) Dar el recorrido en postorden

10. Construir una función imprimeInverso que imprima los elementos de una lista enlazada de enteros en orden inverso a partir de una posición p.

11. Una ordenación de selección recorre un arreglo buscando el elemento más pequeño del mismo. Cuando encuentra el elemento más pequeño éste es intercambiado por el primer elemento del arreglo.

El proceso se repite para el subarreglo que comienza con el segundo elemento del arreglo. Cada pasada del arreglo resulta en un elemento colocado en su posición correcta.

Para un arreglo de n elementos, deberán hacerse n-1 pasadas, y para cada subarreglo deberán hacerse n-1 comparaciones. Cuando el subarreglo bajo proceso contenga un solo elemento, el arreglo habrá quedado terminado y ordenado. Escriba la función selectSort().

12. Se dispone de un árbol binario de elementos de tipo entero. Escriba funciones que calculen:
 - a) La suma de sus elementos.
 - b) La suma de sus elementos que son múltiplos de 3
13. Escribir una función recursiva calcular la cantidad de maneras diferentes en las cuales un entero **N** se puede expresar como la suma de dos enteros menores **p<N** y **x<N** tales que **N=p+x**
14. Implemente una función que tome como entrada un árbol T y un nodo n en el árbol T. Como resultado devuelve el nodo anterior al nodo n del árbol T en un recorrido inorden. Cero si no existe el nodo anterior.
15. Hacer un método que reciba una lista y una pila, el método deberá modificar la lista original, eliminando las posiciones indicadas por cada nodo de la pila.

Lista → 2 → 4 → 6 → 8 → 9 → 3

Pila → 2 → 6

Lista nueva → 2 → 6 → 8 → 9