

CI0112 Programación 1
Segundo examen parcial
Grupo#06
3 horas

P1. Usando una **lista simple**, implemente el método ***boolean es_Lista_Palindrómica()*** que permite saber si el primer elemento en la lista es igual al último, el segundo al penúltimo y así por el estilo. Solo si existe dicha equivalencia entre cada par de elementos, se puede decir que si es listaPalindrómica, de otra forma habría que decir que no lo es

Aspectos puntuables:

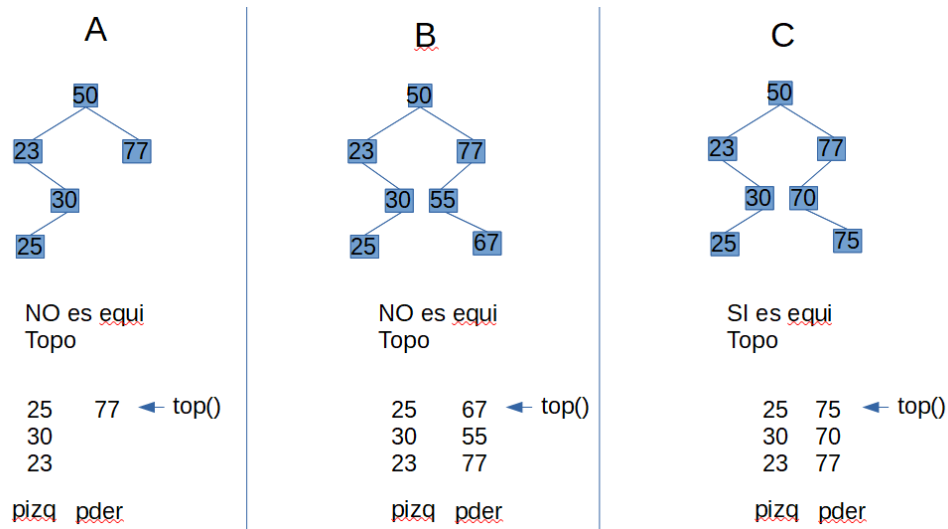
p1.1. implementación del método recursivo ***boolean es_Lista_Palindrómica()*** 25%

p1.2. implementación del controlador que instancia, ejecuta el método y muestra el resultado 10%

P2. Dado el código básico que agrega elementos en un árbol binario (adjunto), usando además 2 pilas: una para valores en la rama izquierda y la otra para valores en la rama derecha, implemente el método ***boolean esEquiTopo()*** que indica si la instanciación actual del árbol es o no equilibrada, concepto bajo el cual vamos a entender:

1. la rama izquierda del árbol coincide con la rama derecha en cuanto a forma
2. el valor del nodo raíz de cada subárbol en la rama izquierda, guarda la misma distancia con la raíz principal que el valor del nodo raíz de cada subárbol complementario en la rama derecha .

Ejemplos:



- a) el arbol A no es equi Topo porque la forma de la rama izquierda no coincide con la forma de la rama de su rama derecha
- b) el arbol B no es equi Topo porque a pesar de coincidir en la forma la rama izquierda y su rama derecha, la distancia entre los nodos con su raíz no es igual, así
 - dist (23,50) → 27 dist (77, 50) → 27 // hasta acá todo bien
 - dist(30, 50) → 20 dist(55, 50) → 5 // ambas diferencias no coinciden NO CUMPLE

- c) el árbol C si es equi Topo porque no solo coincide la forma, sino que los valores correspondientes guardan la misma distancia con el valor raíz, según se muestra usando las pilas:
- $\text{dist}(25, 50) \rightarrow 25$ $\text{dist}(75, 50) \rightarrow 25$ // hasta acá todo bien
 - $\text{dist}(30, 50) \rightarrow 20$ $\text{dist}(55, 70) \rightarrow 20$ // hasta acá todo bien
 - $\text{dist}(23, 50) \rightarrow 27$ $\text{dist}(77, 70) \rightarrow 27$ // hasta acá todo bien

Aspectos puntuables:

p2.1. despliegue de valores en la pila (control interno de uds) 10%

//además, deben crear métodos distintos para llenar los valores de cada pila

p2.2. método para llenar cada pila 10%

p2.3. implementación del controlador que instancia y ejecuta el método 5%

p2.4. implementación del método ***boolean esEquiTopo()*** 15%

P3. Dado el código básico que agrega elementos en un árbol binario (adjunto), usando recursividad, implemente el método ***boolean esEquiTopo(Arbolbin a, int vr)*** que indica si la instanciación actual del árbol es o no equilibrada, concepto bajo el cual vamos a entender:

1. la rama izquierda del árbol coincide con la rama derecha en cuanto a forma
2. el valor del nodo raíz de cada subárbol en la rama izquierda, guarda la misma distancia con la raíz principal que el valor del nodo raíz de cada subárbol complementario en la rama derecha .

Observación: este método es extrañamente recursivo porque debe aplicarse simultáneamente a dos árboles binarios, el actual y el que se recibe como parámetro. Noten que también se recibe un segundo parámetro, que corresponde al valor raíz del árbol.

Aspectos puntuables:

p3.1. implementación del método recursivo ***boolean esEquiTopo(Arbolbin a, int vr)*** 25%

p3.2. implementación del controlador que instancia, ejecuta el método y muestra el resultado 10%