

1.- Al realizar el recorrido de un grafo es necesario ir marcando los nodos visitados, excepto si el grafo es conexo y acíclico

Verdadero

2.- En el TAD Partición, en la representación de boques de árboles, no conviene utilizar al mismo tiempo la unión por altura y la compresión de caminos

Verdadero

3.- Las operaciones del TAD árbol binario permiten insertar y eliminar hojas y nodos internos

Falso

4.- Al insertar en un Árbol B, si el nuevo elemento no cabe en el nodo que le correspondería, se divide el nodo en dos y se promociona un elemento al nodo padre, y en este caso, se permite que exista algún nodo con un solo elemento, independientemente del mínimo permitido según el orden del árbol.

Verdadero

5.- Si implementáramos un TAD Conjunto mediante un ABB, podríamos garantizar que la operación de pertenencia de un elemento a un conjunto siempre sea de coste logarítmico

Falso

6.- No ha sido fácil, pero en la ultima representación del TAD Partición, por fin hemos conseguido al mismo tiempo garantizar que tanto la operación de unión como la de encontrar estén en  $O(1)$ .

Falso

7.- El cálculo de la altura de un árbol es de  $O(n)$ , excepto para los AVL, en cuyo caso el orden del cálculo de la altura es logarítmico

Verdadero

8.- Es cierto que todos los AVL son ABB, pero no es cierto que algunos ABB no sean AVL

Verdadero

9.- No es necesario que Kruskall verifique que no se producen ciclos en la solución, pues al pertenecer los nodos unidos por la arista seleccionada al mismo árbol, ello simplemente no es posible

Falso

10.- Si no pusiéramos la función suma en los algoritmos de Floyd o Dijkstra, en el fondo no pasaría nada, pero aporta legibilidad al código y eso es muy importante

Falso

11.- Sea un árbol binario implementado mediante una representación vectorial. La destrucción del árbol binario debe eliminar los nodos uno a uno es postorden

Verdadero

12.- Siempre que tengamos un origen o un destino definido, debemos usar Dijkstra o Dijkstra inverso en vez de Floyd, por cuestiones de eficiencia, a la hora de calcular el árbol de expansión mínimo.

Verdadero

13.- En el algoritmo de eliminación de una clave en un árbol B, si dicha clave no está en una hoja, se sustituye por su sucesor más inmediato, que se encontrará siempre en el hijo derecho del nodo que contiene la clave a eliminar

Falso

14.- La inserción en el mismo orden de un conjunto de elementos en un ABB y un AVL daría como resultado árboles de la misma altura

Falso

15.- Un recorrido en anchura desde un vértice origen permite encontrar el conjunto de vértices de un componente conexo

Falso

16.- En un subárbol de un árbol cualquiera se cumple la siguiente propiedad: en cada nodo la suma de su altura y profundidad es constante y coincide con la profundidad de la hoja más profunda

Falso

17.- Supongamos un ABB con el elemento x en una hoja cuyo padre tiene el valor y. Entonces, y es el menor elemento mayor que x o bien, y es el mayor elemento menor que x.

Verdadero

18.- Los elementos de un APO se obtienen en orden mediante la extracción sucesiva de estos.

Verdadero

19.- En Prim, en caso de empate entre dos aristas candidatas, es decir, del mismo coste y ambas válidas, debe escogerse la que consiga unir más nodos

Falso

20.- Ni Dijkstra ni Floyd funcionan correctamente con costes negativos, al contrario que Prim y Kruskal.

Falso

21.- Prim y Kruskal no devuelven el mismo resultado siempre, excepto en el caso que haya empates entre diferentes árboles generadores mínimos.

Verdadero

22.- La propiedad de equilibrio de un AVL permite encontrar un elemento en un tiempo de  $O(\log n)$  en el caso peor

Verdadero

23.- La representación vectorial de posiciones relativas es adecuada solamente para arboles parcialmente ordenados

Falso

24.- Para conseguir que la anchura del árbol B sea menor, me interesa crear nodos con el mayor tamaño posible.

Verdadero

25.- En la implementación vectorial de árbol binario, que tiene como invariante que se colocan todos los elementos al principio del vector, la inserción es de coste  $O(1)$ , pero el borrado de coste  $O(n)$

Falso

26.- Un AVL es ABB y el recíproco es cierto.

Falso

27.- La propiedad de equilibrio de un AVL no implica que su altura sea la mínima posible

Verdadero

28.- En la representación por bosque de árboles del TAD Partición, dado que la función encontrar lo que recorre es el camino hasta la raíz, es en el caso peor de orden logarítmico

Verdadero

29.- Todo APO min-max cumple estrictamente con las condiciones que hemos definido para un APO, pero el recíproco no es cierto

Verdadero

30.- La eficiencia espacial de la representación de un árbol binario mediante un vector de posiciones relativas será mayor cuantos más nodos falten en el nivel inferior

Verdadero

31.- Es necesario marcar los nodos visitados en el recorrido de un grafo para evitar provocar ciclos en el mismo

Verdadero

32.- Árbol de expansión de coste mínimo solo hay uno, eso sí, arboles de expansión en general (que no sean de coste mínimo), por supuesto hay muchos

Verdadero

33.- Se podría optimizar el algoritmo de Dijkstra usando un APO, en particular a la hora de buscar el nodo más cercano al origen que todavía no ha sido usado para mejorar a los demás

Verdadero

34.- En un AVL cada nodo tiene un factor de equilibrio de 1 0 o -1 y ello significa que todas las hojas se van a encontrar en el último nivel o en el penúltimo

Verdadero

35.- En un grafo no dirigido, los resultados devueltos por Dijkstra y Dijkstra inverso aplicados al mismo nodo como origen y destino son iguales

Verdadero

36.- La propiedad de orden parcial de un APO implica que siempre va a estar equilibrado

Verdadero

37.- Una cola con prioridad mediante un APO permite eliminar el elemento prioritario con un coste  $O(1)$  en el caso peor

Verdadero

38.- La profundidad del nodo más profundo del árbol es la altura de dicho árbol menos 1

Falso

39.- A partir de los recorridos en PreOrden e InOrden de un árbol general es posible reconstruir el árbol original, si además se conoce el grado del árbol.

Falso

40.- La representación del TAD Árbol general mediante listas de hijos es más ineficiente cuanto más alto es el árbol.

Falso

Nota sacada: 3

Es decir, 26 aciertos y 14 errores.