



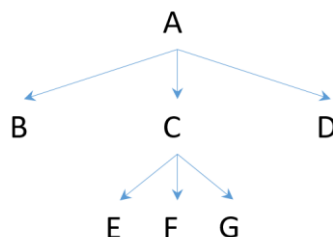
Nombre y apellidos:

DNI:

- Duración del examen: 2 horas y 30 minutos.
- Todas las hojas entregadas deben tener nombre y DNI del alumno y Nº de página/Total de páginas.
- Las respuestas a los ejercicios deberán escribirse en pseudocódigo y estar claramente justificadas, es decir, acompañadas de una breve y clara explicación.
- En todas las preguntas pueden incluirse operaciones auxiliares, debidamente justificadas, si se considera necesario.
- En el apartado a) de las preguntas 2 y 3 deben explicarse claramente las operaciones básicas del TAD (tipos de entrada y salida de las mismas y ecuaciones de definitud) y describir su funcionamiento.

(3 puntos) Ejercicio 1.-

- a) **(0,5 puntos)** Considerando un montículo de máximos vacío (orden alfabético), representado por un vector, indicar el contenido del montículo después de insertar los siguientes elementos: D, A, E, B, I, H, C, y G.
- b) **(0,5 puntos)** Considerar un montículo de mínimos de cadenas de caracteres (orden alfabético), inicialmente vacío, en el que se han introducido los siguientes nombres: Madrid, Guadalajara, Cuenca, Toledo, Albacete, Segovia, Soria, Zaragoza, Badajoz y Valladolid.
- b.1 Si se escribe la raíz del montículo y, a continuación, se elimina el elemento menor y se repiten estas operaciones hasta que el montículo esté vacío, indicar el orden en el que se escribirán las ciudades.
- b.2 Indicar el contenido del montículo, representado por un vector, justo después de eliminar los menores hasta Badajoz, incluido.
- c) **(0,25 puntos)** Dado el siguiente árbol general:



Responder a las siguientes cuestiones (las contestadas erróneamente restan 0,08 puntos)

- c.1 ¿Es homogéneo? (Sí / No)
- c.2 ¿Es semicompleto (o, lo que es lo mismo, casi completo)? (Sí / No)
- c.3 ¿Cuál es su grado? _____



Nombre y apellidos:

DNI:

- d) **(0,25 puntos)** Escribir el resultado de recorrer el árbol anterior en modo pre-orden.
- e) Dada la secuencia de enteros: 20, 50, 45, 35, 70, 19, 40, 34, 37:
- e.1 **(0,25 puntos)** Representar gráficamente el árbol binario de búsqueda resultante. Borrar el 45 en el árbol creado, de forma que siga siendo un árbol binario de búsqueda.
- e.2 **(0,5 puntos)** Representar gráficamente el árbol AVL correspondiente, explicando y realizando los giros que sean necesarios.
- f) **(0,75 puntos)** Partiendo de la especificación del TAD ÁRBOLES_BINARIOS[ELEMENTO] y suponiendo conocida la operación *niveles*: *a_bin* -> *natural* (que devuelve la profundidad de un árbol), escribir la operación **es_completo?** *a_bin* -> **bool** (que devuelve verdadero si el árbol binario es completo y falso si no lo es).

(3,5 puntos) Ejercicio 2.-

- a) Dar la especificación del TAD ÁRBOLES_BINARIOS[ELEMENTO] **(0,25 puntos)**

Partiendo únicamente de la especificación del apartado a), escribir las siguientes operaciones:

- b) *es_estriicto_bin*: *abin* → *bool*, comprueba si un árbol binario es estrictamente binario, es decir, que todo nodo no hoja tiene dos hijos. **(1 punto)**
- c) *nivel_k*: *abin* → lista, devuelve una lista con los elementos del nivel *k*-ésimo de un árbol binario. **(1 punto)**
- d) *es_imagen_espec*: *abin abin* → *bool*, comprueba si un árbol binario es la imagen especular de otro. **(1,25 puntos)**

(3,5 puntos) Ejercicio 3.-

- a) Dar la especificación del TAD ÁRBOLES[ELEMENTO] **(0,5 puntos)**

Partiendo únicamente de la especificación del apartado a), escribir las siguientes operaciones:

- b) *iguales*: *agen agen* → *bool*, comprueba si dos árboles generales son iguales. **(1,25 puntos)**
- d) *es_imagen_espec*: *agen agen* → *bool*, comprueba si un árbol general es la imagen especular de otro. **(1,75 puntos)**