

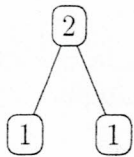
Estructuras de Datos y Algoritmos

Grados en Ingeniería Informática

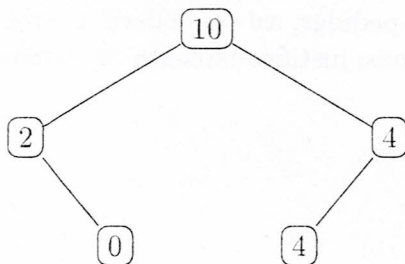
Examen Segundo Cuatrimestre, 2 de julio de 2019. Grupos B y D

1. (3.5 puntos) Un nodo interno de un árbol binario de enteros se dice que es *sumativo* si su valor es igual a la suma de los valores de sus descendientes.

A modo de ejemplo, un árbol vacío o un árbol hoja tendrán 0 nodos internos sumativos. El árbol



tendrá un nodo interno sumativo. Y el árbol



tendrá dos nodos internos sumativos.

Implementa la función

```
unsigned int numNodosSumativos (Arbin<int> a)
```

que devuelva el número de nodos internos sumativos que tiene el árbol dado. Determina así mismo de forma justificada la complejidad de dicha función.

2. (6.5 puntos) Nos han encargado implementar un módulo para la gestión de surtidores en una gasolinera. La gasolinera tiene surtidores, cada uno de los cuáles puede servir distintos tipos de combustible. Cuando un vehículo llega a la gasolinera, se pone en la cola de espera de alguno de los surtidores, y reposta combustible (los detalles de pagos no se cubrirán en este módulo). Así mismo, en cualquier momento un vehículo puede abandonar la cola de espera, y seguir su camino. La implementación de este módulo se llevará a cabo como un TAD *GestorSurtidores* con las siguientes operaciones:

- **crea:** Creación de un nuevo gestor de surtidores.
- **an_surtidor(id_surtidor):** Se añade un nuevo surtidor, con identificador *id_surtidor*, al sistema. Si el surtidor ya existe en el sistema, se señala un error (mediante la excepción *ESurtidorDuplicado*).
- **carga(id_surtidor, tipo_combustible, num_litros):** Carga *num_litros* de combustible *tipo_combustible* en el surtidor *id_surtidor*. Si el surtidor no existe en el sistema, se señala un error (excepción *ESurtidorNoExiste*).
- **pon_en_espera(id_vehículo, id_surtidor):** Pone en espera el vehículo *id_vehículo* en el surtidor *id_surtidor*. Si el vehículo ya está esperando en otro surtidor, o bien *id_surtidor* no existe, se señala un error (excepción *ELlegadaVehiculo*).

- `vende(id_surtidor, tipo_combustible, num_litros) -> resul`: Actualiza el sistema tras realizar una venta de `num_litros` de combustible `tipo_combustible` al primer vehículo que está esperando en el surtidor `id_surtidor`. Tras la venta, se actualiza la reserva de combustible del surtidor, y el vehículo abandona la gasolinera. Una vez realizada la venta, la operación devuelve un objeto que contiene los siguientes campos: (i) el identificador del vehículo al que se ha realizado la venta; y (ii) la cantidad de combustible de tipo `tipo_combustible` que queda aún en el surtidor. Para que esta operación puede realizarse: (i) el surtidor debe existir; (ii) debe haber vehículos esperando para abastecerse en el mismo; (iii) el surtidor debe disponer de la cantidad suficiente de combustible solicitado; y (iv) `num_litros` debe ser mayor que 0. Si se viola alguna de estas condiciones, la operación señala un error (excepción `EErrorVenta`)
- `abandona(id_vehiculo)`: El vehículo `id_vehiculo` abandona la cola en la que está esperando, y sale de la gasolinera. Si el vehículo no está esperando en ninguna cola, la operación no tiene efecto.

Dado que este módulo es un sistema crítico, debe elegirse una representación que permita implementar lo más eficientemente posible las operaciones pedidas, así como llevar a cabo dicha implementación. Para cada operación debe indicarse, además, justificadamente su complejidad.



Surtidores
 tipo combustible
 diesel litros
 gasolina litros