

Estructuras de Datos y Algoritmos

Grados en Ingeniería Informática

Examen FINAL JUNIO, 31 de mayo de 2018

1. (4 puntos) Dado un vector v de longitud n que sólo contiene ceros o unos, queremos saber la longitud de la subsecuencia más larga que no contiene más de m ceros.

Especifica, diseña e implementa un algoritmo iterativo que resuelva el problema anterior de manera eficiente. Escribe el invariante y función de cota que permitan demostrar la corrección del algoritmo implementado. Por último, calcula y justifica el coste del algoritmo conseguido.

Entrada							Salida
n	m	v					
5	0	1	1	1	1	1	5
5	0	0	1	1	0	1	2
5	1	0	1	1	0	1	4
5	2	0	1	1	0	1	5
5	2	0	0	0	0	0	2
5	2	1	0	0	1	0	4
0							

2. (2 puntos) En el desarrollo de un juego de aventuras conversacionales se está representando el mapa del juego mediante un árbol binario de personajes. En cada nodo del árbol hay un personaje (*monstruo*, *caballero* o *dama*) representado a través de un carácter ('M', 'C' y 'D', respectivamente)

Se dice que un nodo está *a salvo* cuando: (i) en él hay una dama; (ii) el número de monstruos que hay en el camino que va desde la raíz a dicho nodo es menor o igual que el número de caballeros que hay en los nodos descendientes de dicho nodo.

Debes implementar una función:

```
int num_a_salvo(const Arbin<char>& mapa);
```

que determine el número de nodos en `mapa` que están *a salvo*. Indica además su complejidad.

3. (4 puntos) Se desea desarrollar una aplicación que permita gestionar un parque natural. En el parque natural hay ecosistemas, cada uno identificado a través de un nombre único. Cada ecosistema contiene un cierto número de ejemplares de cada una de las distintas especies que en él habitan. En un ecosistema no pueden habitar dos especies con igual identificador, aunque una misma especie (igual identificador) sí puede habitar en dos o más ecosistemas distintos. Se pueden añadir nuevos ecosistemas en el parque; en un ecosistema se puede añadir una nueva especie con un cierto número de ejemplares o incrementar el número de ejemplares de una especie ya existente en el ecosistema; se puede obtener un listado ordenado alfabéticamente de las especies que habitan en un ecosistema; se puede consultar el número de ejemplares que hay de una determinada especie en un cierto ecosistema así como el número de ejemplares de una determinada especie que hay en el parque.

Se pide implementar un TAD `ParqueNatural` que proporcione las siguientes operaciones:

- `crea()`: Crea un parque natural vacío.

- **an_ecosistema(ecosistema)**: Añade un nuevo ecosistema con identificador **ecosistema** al parque. Si el ecosistema ya existe, levanta una excepción **EEcosistemaDuplicado**.
- **an_ejemplares(ecosistema,especie,n)**: Añade **n** ejemplares de la especie con identificador **especie** al ecosistema con identificador **ecosistema**. Si **ecosistema** no existe, levanta una excepción **EEcosistemaNoExiste**. Si la especie ya habita en el ecosistema se incrementará su número de ejemplares en **n**; si la especie no habita en el ecosistema, se registrará en el ecosistema esa nueva especie con ese número de ejemplares.
- **lista_especies_ecosistema(ecosistema)->lista**: Devuelve una lista, ordenada alfabéticamente, con los identificadores de las especies que habitan el ecosistema de identificador **ecosistema**. Si **ecosistema** no existe, levanta una excepción **EEcosistemaNoExiste**.
- **numero_ejemplares_en_ecosistema(ecosistema,especie)->numero**: Devuelve la cantidad de ejemplares de la especie **especie** que habitan el ecosistema **ecosistema**. Si **ecosistema** no existe, levanta una excepción **EEcosistemaNoExiste**. Si la especie no habita en el ecosistema, la operación devolverá **0** ejemplares.
- **numero_ejemplares_en_parque(especie)->numero**: Devuelve el número de ejemplares que hay de la especie **especie** en el parque. Si la especie no habita en el parque, la operación devolverá **0** ejemplares.

Debes elegir la representación del TAD teniendo en cuenta que debes conseguir que la implementación de las operaciones sea lo más eficiente posible. Debes, así mismo, indicar justificadamente la complejidad de cada una de estas operaciones.