

Todas las preguntas tienen la misma ponderación (25%).

1. En primer lugar, escribir un TAD *Letra* que incluya, por un lado, todos los constructores necesarios para representar cada letra (mayúscula), y, por otro lado, las operaciones *esVocal* y *esConsonante*.

En segundo lugar, partiendo de un TAD *Pila* de elementos de tipo *Letra* con las operaciones *crear*, *esVacía*, *pop*, *tope* y *push*, añadir a su especificación formal algebraica las siguientes operaciones:

- a) *quitarVoc*: dada una pila de letras, devuelve una pila que solo contenga las consonantes de la pila original en el mismo orden.
- b) *quitarCon*: dada una pila de letras, devuelve una pila que solo contenga las vocales de la pila original en el mismo orden.
- c) *agrupar*: dada una pila de letras, devuelve una pila con las vocales de la pila original al fondo y las consonantes de la pila original encima, en el mismo orden.

Si es necesario, se pueden añadir otras operaciones.

2. ¡Ya queda muy poco para que empiece el mundial de fútbol! Una empresa de apuestas ha planteado el siguiente mecanismo. Cada día se juegan entre 1 y 4 partidos del mundial. Los usuarios deben decir un día (el que quieran) y predecir cuántos goles marcará cada equipo ese día (por ejemplo, si solo hay un partido, A contra B, deben predecir los goles de A y de B; y si hay 4 partidos, deben predecir los goles de los 8 equipos, en orden). Así que cada apuesta contendrá información sobre el usuario que la hace (un identificador numérico único), el número de día de la apuesta (desde 1 hasta 30), y los goles predichos (2, 4, 6 u 8 números enteros no negativos, según el día).

Como se esperan millones de usuarios y miles de millones de apuestas, la empresa quiere optimizar la búsqueda de los ganadores de cada día. Es decir, una vez que ha pasado un día y se saben los resultados, conocer rápidamente quiénes son los usuarios acertantes que se reparten el premio de ese día.

Escribir una solución eficiente para el problema anterior usando tablas de dispersión. Hay que detallar todos los aspectos de la solución: qué tipo de tablas se usan, cómo se almacena la información, qué función de dispersión se usa, cómo es la operación de inserción, cómo es la de consulta, qué tamaño de tabla se usa, qué estrategia de reestructuración, etc.

3. Resolver el problema anterior usando la estructura de datos basada en árboles que consideres más adecuada. Recuerda que el objetivo es conseguir que las operaciones de inserción de una apuesta nueva y consulta de las apuestas ganadoras de un día sean muy eficientes. Debes igualmente detallar todos los aspectos de la solución y mostrar algún ejemplo gráficamente.
4. Tenemos que diseñar un plan de evacuación de la Facultad. Para cada ubicación (aula, laboratorio, despacho, rellano...) nos piden determinar la forma más rápida de abandonar el edificio por alguna de sus salidas.

Supongamos que disponemos de n ubicaciones y, entre ellas, un subconjunto de m salidas. Determinar el tipo de grafo que modela el problema, el significado de los nodos y de las aristas, y escribir un algoritmo que determine de forma eficiente, para cada ubicación, la vía de escape más rápida (más corta) para escapar del edificio. Explicar bien la idea e indicar el orden de complejidad del algoritmo resultante.