

Tipos Abstractos de Datos

Práctica 1 TAD Bolsa

Objetivo

El objetivo de esta actividad es que el alumno refuerce su conocimiento sobre el tema de Tipos Abstractos de Datos (TAD) y sus habilidades de programación.

Descripción general

La práctica consiste en implementar un tipo abstracto de datos para trabajar con una colección de elementos llamada Bolsa. Los elementos de una bolsa no tienen un orden definido y no importa si se repiten.

Material

El material de esta práctica consta de los siguientes archivos:

- **InterfazBolsa.class** interfaz para el tipo abstracto de datos Bolsa.
- **Bolsa.java** clase que implementará la interfaz `InterfazBolsa`.
- **PruebaBolsa.class** programa para probar la clase `Bolsa`.
- Documentación:
 - `InterfazBolsa.html` documentación de la interfaz `InterfazBolsa`.

Desarrollo

1. Descargar los archivos `InterfazBolsa.class`, `Bolsa.java` y `PruebaBolsa.class` en el directorio donde se va a desarrollar la práctica.
2. Leer la documentación y revisar el material de la práctica. Recordar que la firma de los métodos proporcionados y de los requeridos en el desarrollo de la práctica no podrán ser modificados.
3. En la clase `Bolsa` se debe implementar la interfaz `InterfazBolsa`. Los métodos por implementar son los siguientes:

- a. **Constructores** incluir uno por omisión que cree una bolsa para 15 elementos, un constructor que reciba un valor entero que indique la capacidad de elementos de la bolsa y un constructor de copia.
 - b. **numElementos** devuelve el número de elementos que hay en la bolsa.
 - c. **agregar** agrega un elemento a la bolsa solo si hay espacio disponible. Devuelve true si la operación fue exitosa o false en otro caso.
 - d. **eliminar** elimina de la bolsa el elemento pasado como parámetro todas las veces que aparece en la bolsa. Devuelve true si la operación fue exitosa o false en otro caso.
 - e. **contiene** devuelve true si la bolsa contiene al menos una vez al elemento indicado como parámetro o false en otro caso. Para ello se hace uso del método equals.
 - f. **cardinalidad** devuelve el número de veces que el elemento aparece en la bolsa.
4. Probar la implementación realizada ejecutando el programa `PruebaBolsa`. Si el programa funciona adecuadamente se verán los siguientes mensajes:

```
Creando una bolsa con los siguientes elementos: { Hola, Hello,
Guten Tag, Aloha, Saluton, Hei, Bonjour, Namaste, Shalom, Halo,
Parev, Kon-nichiwa, 7, 13, Aloha }
```

```
La bolsa recién creada tiene 15 elementos.
```

```
¿La bolsa contiene el numero 13? true
Eliminando el elemento numero ... 13
¿La bolsa aun contiene el numero 13? false
La bolsa ahora contiene solamente 14 elementos.
```

```
¿Cual es la cardinalidad de la palabra "Hi"? 0
¿Cual es la cardinalidad de la palabra "Aloha"? 2
```

```
Eliminando el elemento ... "Aloha"
Ahora la bolsa tiene 12 elementos.
```

5. Escribir una tabla con la complejidad de los métodos del TAD para la colección Bolsa que implementa la interfaz `InterfazBolsa`.