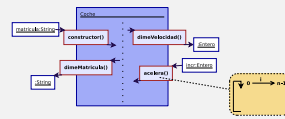


Grado en Ingeniería de Sistemas de Telecomunicación, Sonido e Imagen



Práctica 3



Escola Politècnica Superior de Gandia

DSIC

Departament de Sistemes Informàtics i Computació

Sesiones

- Grupos martes: nov-13, nov-20, nov-27
- Grupos viernes: nov-16, nov-23, nov-30



Práctica 3

Objetivos

- Implementación de una clase dado su diseño.
- Uso de sentencias condicionales y bucles.
- Uso de arrays.
- Utilización de clases ya existentes.

¡ Atención !

- ▷ Se recuerda que las prácticas deben prepararse antes de acudir al aula informática. Esto incluye leer el código proporcionado y probarlo.
- ▷ La realización de las prácticas es un trabajo individual y original. En caso de plagio se excluirá al alumno de la asignatura. Por tanto es preferible presentar el trabajo realizado por uno mismo aunque éste tenga errores.



1

Utilización de la clase de biblioteca ArrayList

Lee y prueba¹ el programa `E1.main()`, que contiene un ejemplo de utilización de la clase de biblioteca `ArrayList`.

Un `ArrayList` es un array mejorado, con la capacidad de cambiar de tamaño en tiempo de ejecución.

Los elementos de un `ArrayList` han de ser objetos, por eso para poder declarar un `ArrayList` de enteros se utiliza la clase `Integer`²:

```
ArrayList<Integer> numeros = new ArrayList<Integer> ();
```

Igualmente, si quisiéramos un `ArrayList` de reales, utilizaríamos la clase `Double`.

```
ArrayList<Double> elementos = new ArrayList<Double> ();
```

¹ Se puede ejecutar específicamente el `main()` de `E1` pulsando el boton derecho del ratón sobre `E1.java` y eligiendo en el menú Run File (Ejecutar fichero).

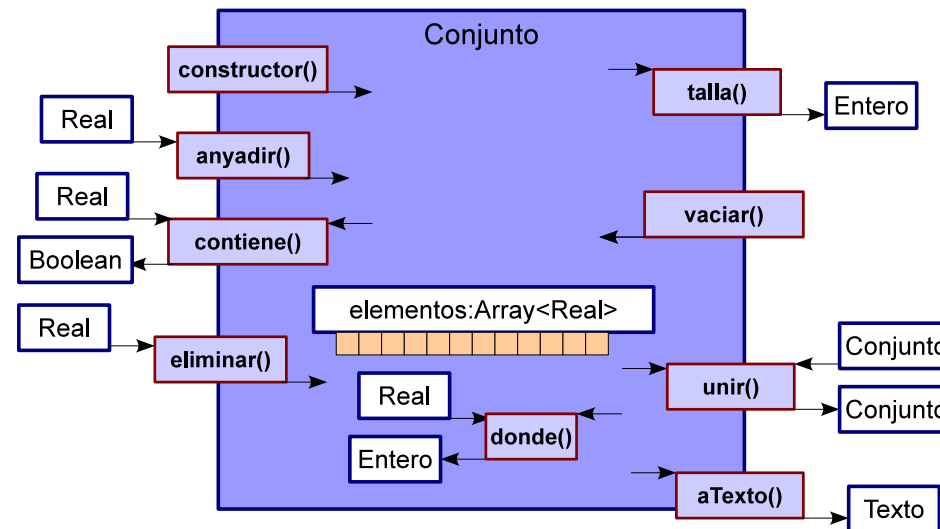
² `Integer` es el envoltorio de un `int`.



2

Implementación del diseño de la clase Conjunto

El diseño que hay que implementar es éste:



donde:

- Las operaciones públicas que tiene un conjunto y que deben implementarse son:
 - un constructor por defecto.
 - `talla()`: devuelve la cantidad de elementos que contiene el conjunto.
 - `anyadir()`: añade un nuevo elemento al conjunto.
 - `contiene()`: devuelve `true` si el elemento dado pertenece al conjunto.
 - `eliminar()`: elimina el elemento indicado.
 - `unir()`: une dos conjuntos, devolviendo uno nuevo.
 - `vaciar()`: vacía el conjunto.
- Los elementos del conjunto se guardan de forma privada en un array de números reales llamado `elementos`.

- Debe haber un método privado llamado `donde()` que dado un elemento diga en qué posición del array se encuentra. Si ese elemento no está contenido `donde()` devolverá la constante `NO_ESTA`.
- Se da implementado un método público `String aTexto()` que devuelve como texto el contenido de un conjunto, para ayudar en las pruebas.

Trabajo a realizar

Hay que implementar la clase `Conjunto` (en `Conjunto.java`) y probarla (en `E2.main()`) tal y como se ha descrito en el diseño.

Para probar la implementación se puede escribir varias funciones `main()` que sin hacer nada en particular comprueben que todos los métodos implementados funcionan correctamente.

La implementación hay realizarla de la siguiente forma:

1. Implementar el constructor.
2. Implementar `vaciar()`.
3. Implementar `talla()`.
4. Probar los métodos anteriores.
5. Implementar `donde()`.
6. Implementar `contiene()` llamando `donde()`.
7. Implementar `anyadir()` llamando a `contiene()`. (Un conjunto no puede tener elementos duplicados).
8. Probar todos los métodos anteriores (incluyendo los primeros).
9. Implementar `eliminar()` llamando a `donde()`.
10. Probar todos los métodos anteriores.
11. Implementar `unir()` llamando a `anyadir()`.
12. Probar todos los métodos anteriores.

Trabajo voluntario valorable

Implementar otros métodos para la clase `Conjunto` como intersección, diferencia, diferencia simétrica, etc.



3

Entregas

3.1

Cuestionario

Se entregará en papel para poder entrar a la primera sesión.

1. ¿Qué hacen las siguientes líneas de código del programa `E1.main()`?

`numeros.add(-40);`

`numeros.add(1, 1234);`

`numeros.set(1, 2345);`

`numeros.set(i, numeros.get(i)*2);`

`numeros.remove(2);`

`numeros.clear();`

2. Escribe un algoritmo, con la notación vista en clase, para averiguar en qué posición de un array se encuentra un valor dado.
3. Escribe el esqueleto³ de la clase `Conjunto`.

³ El código de la clase sin la implementación de los métodos

3.2

Código completo de la práctica

El fichero `p3.zip` con el código completo se debe entregar en la tarea de esta práctica en poliformaT antes de la revisión oral con el profesor correspondiente y en todo caso [hasta el 30 de noviembre](#).



6 novembre 2012