

Problema 4

Escribir un programa para gestionar una clase llamada **ListaAyudas** que contiene una lista de alumnos que solicitan una ayuda, evitando la posibilidad de insertar alumnos repetidos

La lista contendrá objetos de la clase **Ayuda**, que contienen un alumno y la ayuda solicitada en euros

- Al constructor se le pasan el alumno y la ayuda solicitada
- Redefine los métodos **equals()** y **hashCode()** para que se basen en el **equals()** y **hashCode()** del alumno
- Define el método **toString()** que convierte la ayuda a **String**

La secuencia de solicitudes es importante, ya que si se agotan los fondos, las solicitudes se atienden en orden de llegada

Problema 4 (cont.)

ListaAyudas tiene operaciones para

- añadir un alumno al final de la lista; lanza la excepción **Repetido** si el alumno ya ha solicitado una ayuda
- saber la ayuda solicitada por un alumno; lanza **NoExiste** si no se encuentra en la lista
- mostrar en pantalla la lista de todas las ayudas

Evaluar la eficiencia de estas operaciones

Hacer un programa principal que mediante un menú permita al usuario usar la clase **ListaAyudas**

¿Cómo podríamos mejorar la eficiencia de la operación de añadir?