Contenido

[Primer trimestre 1](#_Toc105408848)

[Ejercicio 1 (5p) 1](#_Toc105408849)

[Segundo trimestre 1](#_Toc105408850)

[Ejercicio 2 (5p) 1](#_Toc105408851)

NOTA: Las personas que solo tengan que pendiente un trimestre solo deberán entregar la parte que tienen suspensa

El ejercicio se entregará con el formato APELLIDOS\_NOMBRE\_ED\_ORDINARIA1.zip

Es necesario entregar el documento Word con capturas de pantalla, el proyecto netbeans y los ficheros de visual paradigm para su corrección.

# Primer trimestre

## Ejercicio 1 (5p)

Dado el código del archivo adjunto, se requiere saber qué hace. Para ello realizar las siguientes tareas mediante las funciones que aporta el entorno de desarrollo Netbeans. 0.5 p cada apartado:

* Dar formato al código – captura de pantalla con la opción adecuada del IDE.
* Mediante depuración, localizar y eliminar el código muerto, incluidos métodos. – Capturas de pantalla del proceso
* Refactorizar nombres de variables y métodos por otros más significativos, es decir, relacionados con su funcionamiento. – captura de pantalla
* Incluir el javadoc de los métodos que queden, incluido el main

Nuestro objetivo final es crear una librería de funciones por lo que debemos:

* Crear la clase utilidades.java dentro del paquete de nuestro proyecto, que será visible desde cualquier clase, y contendrá únicamente el código funcional del ejercicio en una función, también visible desde cualquier clase. 1p
* Crear una clase de prueba (pruebas unitarias) que demuestre vuestra teoría de funcionamiento del programa. Se deben probar al menos dos casos válidos y uno no válido. 1p
* El método main constará de: 1p
  + Recepción de datos de entrada para la función
  + Muestra del resultado de la ejecución de la función
  + Mensajes para que el usuario entienda la ejecución del programa

# Segundo trimestre

## Ejercicio 2 (5p)

Se desea modelar una aplicación informática que represente el funcionamiento de un cajero automático. Para simplificar el ejercicio desde el cajero solo se podrán realizar 3 operaciones básicas: ingresar y retirar dinero y cambio de PIN.

Se deberá pensar una estructura lógica y coherente del programa que implicará:

* Identificación, a criterio del alumno/a las clases necesarias (mínimo tres clases básicas, máximo ocho)
* Atributos y métodos de esas clases
* Relaciones entre esas clases
* Funcionamiento e interacciones posibles del programa

Se pide:

1. **Creación del diagrama de clases en Visual Paradigm** 
   * + Identificación de clases, atributos y métodos: 1 p
     + Identificación de relaciones entre clases: 1 p
     + Correcta asignación de cardinalidades entre relaciones: 0,25 p
2. **Creación del diagrama de casos de uso en Visual Paradigm** 
   * Coherencia con respecto al diagrama de clases anteriormente creado 0,5p
   * Correcta elección de actores, asociaciones y correcta participación en el diagrama. 1,25p
3. **Explicación justificada y argumentada de ambos diagramas 1p**