

## Práctica 5

### Pruebas de código Orientado a Objetos

#### Objetivos:

- Aplicar los métodos para la generación de casos de prueba de código orientado a objetos incluyendo: pruebas de métodos, pruebas de clases y pruebas de herencia.

#### Descripción del proyecto

El proyecto simula el comportamiento de un sistema de subastas que viene representado en la Figura 1. Un **usuario** registrado (que se caracteriza por el *nombre* y el *crédito* disponible) puede dar de alta una subasta para vender un producto determinado. Este usuario es el propietario de la subasta. En ese instante, la subasta pasará a estar en estado “abierta” (ver Figura 2), estado en el que se podrá pujar por el producto.

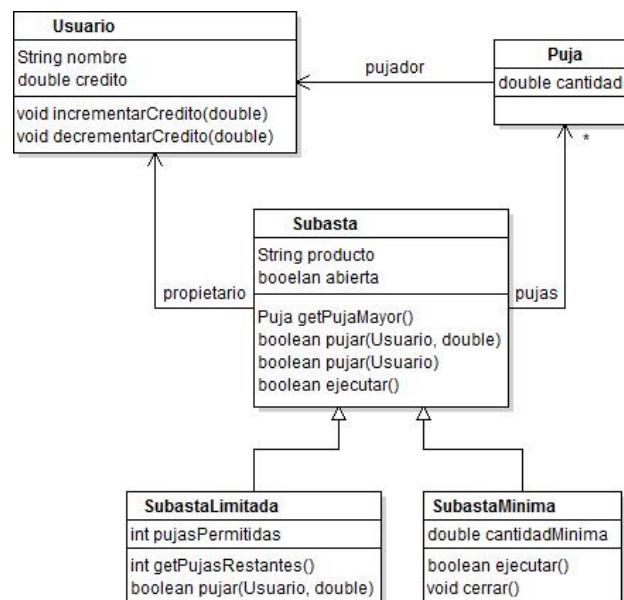


Figura 1 Diagrama de clases del sistema de subastas.

Una **puja** se caracteriza por el usuario que realiza la puja y la cantidad por la que se puja. Para poder **pujar** es un requisito que la subasta esté abierta, en ese caso se debe cumplir que el usuario que realiza la puja sea distinto del propietario de la subasta, que la cantidad por la que se puja sea mayor que la última puja realizada (puja mayor) y que el usuario que realiza la puja tenga crédito suficiente. En el caso de cumplir estas condiciones se registrará la puja en la subasta. Al pujar se puede obviar la cantidad por la que se puja, en cuyo caso se entiende que se está pujando por un euro más que la puja mayor.

En cualquier momento, el propietario podrá **ejecutar** la subasta, con lo que la subasta pasará a estar en estado “cerrada” (ver Fig. 2). Ejecutar la subasta significa realizar la transferencia de crédito del usuario que ha ganado la subasta (el usuario que ha realizado la puja mayor) al propietario de la subasta (que recibe el dinero por el producto subastado).

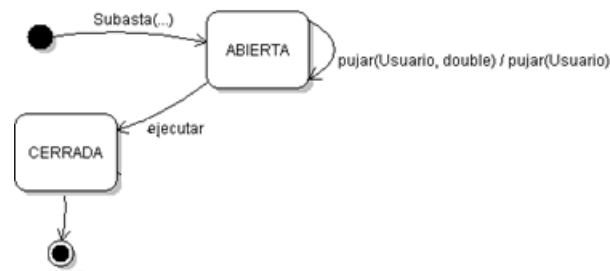


Figura 2 Diagrama de estados de un objeto de tipo subasta.

Existen dos tipos de subastas:

- **Subasta limitada:** es un tipo de subasta que se caracteriza porque sólo admite un número máximo de pujas fijado en el constructor. Una vez que se alcanza el número máximo de pujas la subasta se ejecuta automáticamente.
- **Subasta mínima:** es un tipo de subasta que se caracteriza porque sólo se podrá ejecutar la subasta si la puja mayor alcanza una cantidad mínima, fijada en el momento de construir la subasta. Puesto que puede ser que nunca se alcance la cantidad mínima fijada, esta subasta se puede cerrar en cualquier momento.

## Ejercicios

1. Descarga el proyecto Eclipse que se proporciona en la Tarea del Aula Virtual y renómbralo convenientemente.
2. Elabora un documento de texto que contenga una tabla con el diseño de los casos de **prueba de métodos** para la clase `Subasta` que aseguren la cobertura del código, dicha tabla debe contener, como mínimo, los siguientes datos:
  - a. Método sobre el que se realiza la prueba.
  - b. Nombre significativo del test propuesto.
  - c. Breve descripción del objetivo del test diseñado
  - d. Configuración y parámetros de entrada al test
  - e. Estado/valor esperado después de la ejecución

**Nota:** Recuerda que en los test OO es importante probar que tras la construcción el estado del objeto creado es correcto

3. Implementa los casos de **prueba de métodos** para la clase `Subasta` en la clase `SubastaTest` diseñados en el paso anterior.
4. Analiza el diagrama de estados de la clase `Subasta` y añade al documento de texto elaborado en el apartado 2 las **pruebas de clase aleatorias** de manera que aseguren que se ejercitan todos los posibles estados de los objetos. La tabla debe contener la siguiente información:
  - a. Nombre significativo del test propuesto.
  - b. Secuencia aleatoria que se pretende probar
  - c. Configuración y parámetros de entrada al test
  - d. Estado/valor esperado después de la ejecución

**Nota:** Si las secuencias aleatorias ya están cubiertas por las pruebas de métodos indica el nombre del test que cubre dicha prueba.

5. Añade la implementación de los casos de **pruebas de clase aleatorias** diseñados en el apartado anterior en la clase `SubastaTest`

**Recuerda:** Si las secuencias aleatorias ya están cubiertas por las pruebas de métodos no es necesario duplicar el caso de prueba.

6. Implementa las clases de prueba `SubastaMínimaTest` y `SubastaLimitadaTest` como descendientes de `SubastaTest`. Ejecuta todos los test de `SubastaTest` sobre objetos de prueba de los subtipos para ver qué casos de prueba hay que redefinir y qué nuevos casos de prueba hay que añadir.
7. Implementa los casos de prueba de cada una de las subclases (`SubastaMínimaTest` y `SubastaLimitadaTest`) siguiendo las recomendaciones vistas en clase.

### Entregables

La entrega se realizará en la tarea del Aula Virtual asignada a la práctica con fecha de entrega 16/12/2022 23:55 horas y se debe entregar:

- Archivos de texto con el diseño de los casos de prueba
- Proyecto Eclipse completo (para ello se debe comprimir la carpeta del proyecto, no el espacio de trabajo Eclipse) a través de la tarea abierta en el Aula Virtual.