

## Instrucciones

- Lee cuidadosamente el examen hasta el final. Algunas decisiones de diseño tomadas para implementar las primeras partes pueden afectar a la complejidad en la implementación de las últimas.
- En el examen deberás partir de la *práctica 5* realizada durante el curso. **En breve se indicará el procedimiento de acceso a dicho código.**
- El código entregado *debe compilar*.
- En la corrección del examen se valorará el funcionamiento, la claridad del código, el uso conveniente de los medios proporcionados por la Programación Orientada a Objetos (herencia, polimorfismo...) y comentarios. Para la evaluación se tendrá en cuenta tanto lo pedido en el examen como el código del que se parte (implementación de la práctica 5).
- Romper la encapsulación de las clases (acceso a atributos privados y protegidos desde clases externas, utilización de atributos públicos, etc.) implica suspender el examen.
- Incluye, además con el código, una descripción de los pasos y modificaciones que has realizado para implementar lo que se pide en el examen. Este informe de cambios puedes hacerlo en papel o adjuntarlo como un fichero de texto con el resto de los archivos de código fuente. En ese caso deja el fichero en “el raíz” del .zip entregado.
- Si hiciste las partes opcionales de la práctica 4 y 5, se valorará que se integren las ampliaciones realizadas.
- La puntuación de cada apartado está calculada *sobre 10 puntos*, aunque supondrá el 40% de la nota final.

## Consejos

Es preferible realizar el examen en el orden siguiente:

1. Implementar cada uno de los apartados anteriores. Tras la implementación de cada parte, comprobar que el código compila y funciona y guardar una copia del mismo. De este modo siempre tendremos algo que entregar que sabemos que compila y que funciona (aunque no esté completo).
2. Revisar y corregir los posibles errores o deficiencias que se arrastren de la práctica base, pues ese código también se evalúa.

## Instrucciones de entrega

- Para entregar la solución al examen, crea un fichero zip. En él debes incluir todo el proyecto una vez limpiado de archivos intermedios, y con el fichero `alumnos.txt` con tu nombre completo.
- Nombra al fichero `NN_Apellido1Apellido2.zip`, donde NN indica el número de grupo (con dos dígitos).
- La estructura del fichero será la misma que la utilizada durante el curso.
- Si optas por escribir la descripción de los pasos y modificaciones realizadas durante el examen en un fichero de texto, inclúyelo en el raíz del archivo comprimido.
- Para entregar el examen, se utilizará el mecanismo de entregas disponible en el laboratorio. En particular, cerca del final del examen se habilitará la unidad U: en la que deberás dejar la solución al examen. Si deseas entregarlo antes, indícaselo al profesor para que habilite la unidad.
- Antes de abandonar el laboratorio debes pasar por el puesto del profesor para asegurarte de que lo que se ve en el puesto del profesor es lo que has entregado y firmar en la hoja de entregas.



## Enunciado

Vamos a ampliar la funcionalidad de la **Práctica 5** incluyendo una tienda en la que el jugador puede usar los puntos conseguidos a lo largo del juego para comprar objetos. La tienda estará disponible en cualquier momento; es decir, no es una habitación especial situada en una localización particular del mapa.

A partir de ahora todos nuestros items tienen un precio que aparecerá en la descripción del item (el precio de los items que hemos manejado durante el año será cero):

```
--item[Agua]=Botella de agua, entran tres tragos ($10)//3
```

Para interactuar con la tienda vamos a añadir dos nuevos comandos a nuestra aventura conversacional:

- **shop**: Muestra una lista con los objetos de la tienda y su precio.
- **buy <nombreItem>**: Usa los puntos acumulados por el jugador para comprar un item (identificado por **nombreItem**) de la tienda. Si el jugador tiene suficientes puntos para comprar ese item, el item desaparecerá de la tienda para incorporarse al inventario de jugador y la puntuación del jugador se reducirá en los puntos que haya costado el item comprado. Si el item no está en la tienda o el jugador no tiene suficientes puntos entonces no se podrá completar la ejecución del comando. Si el jugador ya tiene un objeto con ese nombre, tampoco se podrá comprar.

El contenido inicial de la tienda se cargará del fichero de mapa.

A continuación aparece la secuencia más razonable en la que implementar la nueva funcionalidad.

## [2 puntos] Parte A

Crea una clase **Shop** con todos los métodos necesarios para poder interactuar con ella. El mapa contendrá una referencia a un objeto de esta clase y será el responsable de crearla. Inicialmente haz que la tienda se construya con una serie de items por defecto. Pon a cada item el precio que quieras.

## [3 puntos] Parte B

Integra el comando **shop** explicado anteriormente *en la versión de consola*. Cuando se ejecute, deberá aparecer la descripción de los objetos puestos a la venta en la tienda, como por ejemplo:

```
The shop contains the following items:  
--item[chuleta]=Fria, dura, y con nervio ($9)//3  
--item[ganzua]=Abre cualquier puerta ($5)  
--item[mechero]=Me gusta el fuego ($2)//1
```

## [2 puntos] Parte C

Haz las modificaciones necesarias para integrar el comando **buy** en la *versión de consola*. Cuando se ejecute correctamente deberá aparecer la información del usuario actualizada. En caso de no poder comprarlo aparecerá el siguiente mensaje:

```
Unable to buy <itemName>
```

## [1 punto] Parte D

Integra el comando `buy` en la *versión de ventana*. Para eso, añade un nuevo botón `Buy` en la botonera que, al pulsarse, compre el objeto cuyo nombre aparece en el *EditBox* que hasta ahora sólo se utilizaba para recoger (*pick*) objetos.

## [2 puntos] Parte E

Haz las modificaciones necesarias para hacer que la tienda se cargue desde el mapa. La tienda aparecerá al final del fichero (justo antes de `EndMap` y después de la sección de carga de items), en una sección con el siguiente formato:

```
...
BeginShop
food 0 chuleta Fria,_dura,_y_con_nervio 10 3 9
key 1 ganzua Abre_cualquier_puerta door 2 5
valuable 2 mechero Me_gusta_el_fuego 30 2
EndShop
```

Observa que la diferencia entre las descripciones de estos items y las utilizadas para describir los objetos de cada habitación es la última parte: el `room <numero>` que indicaba en qué habitación aparecía el objeto ha sido sustituida por un número que indica su precio de venta.