



# TERCERA PRÁCTICA

## *Programación*

### *Curso 2006-2007*

#### *Ingeniería en Informática*

#### *Universidad Carlos III de Madrid*

## 1. Instrucciones generales

- Durante este curso se deberán realizar **tres prácticas**, cuyas fechas de entrega se pueden encontrar en la página Web de la asignatura:  
(<http://galahad.plg.inf.uc3m.es/~docweb/pr-inf/>)
- Las prácticas se realizarán en Java, usando la máquina virtual de Java 2 (versión 1.5) y el entorno de desarrollo gráfico Eclipse (versión 3.1.1)
- Estas prácticas deberán realizarse **obligatoriamente en grupos de dos** alumnos.
- Las **tres prácticas están relacionadas** entre sí: la segunda es continuación de la primera y utiliza el código creado para ésta, y a su vez la tercera es continuación de la segunda. No obstante, para facilitar la realización de las prácticas y homogeneizar los resultados, **después de la entrega de la primera práctica** los profesores pondrán a disposición de los alumnos, en la página Web de la asignatura, el **código Java que se debe usar como punto de partida en la segunda práctica**. De igual forma, después de la entrega de la segunda práctica, se publicará el código Java de partida de la tercera.
- **Se puede entregar una práctica aunque no se haya entregado la anterior**, partiendo siempre del código Java publicado por los profesores.
- Cada práctica se calificará con un máximo de **un punto (1 punto)**.
- Las prácticas tendrán una **parte obligatoria** y **partes opcionales** que servirán para **subir nota**. Si un alumno realiza solamente la parte obligatoria podrá obtener la nota máxima, aunque no haya realizado ninguna de las partes opcionales. La nota máxima será en cualquier caso 1 punto.

## 2. Práctica tercera

### 2.1 Introducción

En esta tercera práctica se va a continuar con la implementación del juego de comecocos (PacMan), añadiendo nuevas funcionalidades a las de la primera y segunda práctica (la posibilidad de jugar al juego en tiempo real, y poder guardar la partida actual). Los alumnos deben partir del **código Java publicado por los profesores**, en **ningún caso** deben utilizar como punto de partida **su propio código** de la primera o segunda práctica, aunque podrán usar partes de él si lo estiman conveniente.

## 2.2 Objetivos docentes de la práctica

Los objetivos docentes de esta práctica son:

- Afianzar los conocimientos adquiridos en las prácticas anteriores.
- Partiendo de un código ajeno proporcionado, entenderlo y completarlo para cumplir ciertos requisitos especificados.
- Poner en práctica los conocimientos sobre ficheros estudiados, para leer y escribir en ellos.
- Entender y poner en práctica la programación orientada a eventos, respondiendo a las teclas pulsadas por el usuario en cualquier momento.

## 2.3 ¿Qué se proporciona?

Para la realización de la práctica 3 se proporciona el siguiente código java:

- Código solución de la práctica 2, que incluye las clases necesarias para jugar al comecocos en modo textual.
- Código para mostrar en pantalla de manera gráfica el juego del comecocos. La clase principal que se proporciona es la llamada **RellenarPorAlumno** que tiene las siguientes características:
  - Un método **main** a rellenar por el alumno.
  - Un método **siguienteTiempo** que invoca el programa cada 500 milisegundos con un código de ejemplo para rellenar el tablero de forma aleatoria.
  - Un método **teclaPulsada** que es invocado por el programa cada vez que el usuario presiona cualquier tecla.
  - Un método heredado **pintarCuadradoTablero** que puede utilizar para seleccionar el tipo de una determinada casilla del tablero.
  - Unas **constantes** heredadas que representan los distintos tipos (iconos) que puede tener una determinada casilla del tablero (variables estáticas públicas que comienzan con **ICONO\_**, heredadas de la clase **Temporizador**).
  - Un método heredado **establecerPuntos** que puede utilizar para establecer el número de puntos acumulados en la partida.
  - Un método heredado **obtenerPuntos** que puede utilizar para obtener el número de puntos acumulados en la partida.
  - Un método heredado **establecerTiempoDeEspera** que permite establecer el tiempo de espera en milisegundos entre llamadas al método **siguienteTiempo**.

## 2.4 ¿Qué se pide?

- Implementación del juego del comecocos en modo gráfico en tiempo real: Utilizando el código proporcionado como punto de partida, entender y modificar la implementación de la clase **RellenarPorAlumno** (concretamente los métodos **siguienteTiempo** y **teclaPulsada**), para poder jugar al juego del Comecocos en modo gráfico. Para ello, se deberán cumplir los siguientes requisitos:
  - Cada vez que pase un determinado tiempo (500 milisegundos), los enemigos se moverán una posición en el tablero de juego.
  - Utilizando la tecla 'q' el usuario podrá salir de la aplicación (ya implementado en el código de ejemplo).
  - Utilizando las teclas 'w', 's', 'a' y 'd' el usuario podrá modificar la dirección actual del protagonista.

- Cada vez que el protagonista se coma un punto, se sumarán 10 puntos al marcador.
- Posibilidad de salvar y cargar el estado actual de la partida: Se deberá modificar la implementación de la clase **RellenarPorAlumno** para cumplir los siguientes requisitos:
  - Al salir del programa pulsando la tecla 'q', se escribirá en un fichero, mediante serialización, el estado actual de la partida de manera que dicho estado pueda ser recuperado posteriormente.
  - Al iniciar el programa, si existe el fichero en el que se guardó el estado de la partida anterior se restaurará dicho estado para continuar con la partida tal cual se dejó.
  - Se tendrá otro fichero distinto, preferiblemente de texto, en el que se guardarán las diez máximas puntuaciones obtenidas. Al terminar una partida se comprobará si la puntuación está entre las diez máximas y en caso positivo se incluirá en el fichero.
  - El código debe gestionar las posibles excepciones que se produzcan al leer o escribir datos en ficheros.

## 2.5 Actividades optativas

- Cada vez que el protagonista se haya comido un total de 5 puntos, incrementar el nivel de juego acelerando el ritmo de manera que el juego se mueva más deprisa (el tiempo de espera entre tiempos será menor).
- Entender y modificar el código de la clase **ComecocosPractica3** para que encima del marcador de puntos se muestre otro marcador con el número de vidas del protagonista. El protagonista tendrá un total de 3 vidas, y cada vez que un enemigo le alcance se decrementará el número de vidas.
- Modificar la interfaz gráfica para que al terminar la partida muestre las diez máximas puntuaciones guardadas en el fichero.

## 2.6 Propuesta de las actividades a realizar en cada sesión

- Todas las sesiones: libres para realizar la práctica 3.

## 2.7 Entrega de la práctica

- La práctica se deberá entregar el **Martes 5 de Junio**.
  - En Leganés se podrá entregar directamente a los profesores en la hora de clase o tutorías o depositar en los casilleros de los profesores de prácticas situados en la entrada principal del edificio Sabatini. El horario límite de recogida de prácticas son **las 20:00 horas**.
  - En Colmenarejo se deberá entregar en mano al profesor como muy tarde al final de la clase de prácticas de ese día. El horario límite de recogida de prácticas son **las 13:00 horas**.
- Se deberá entregar una **memoria explicativa** de los desarrollos realizados, que en general no debe contener listados de código, salvo los necesarios para la correcta explicación del mismo.
  - La memoria constará de:
    1. Portada: con el nombre, apellidos, número de alumno, grupo (81, 82, 83) y campus (Leganés o Colmenarejo) de cada uno de los alumnos pertenecientes al grupo.
    2. Índice: tabla de contenido del documento.
    3. Manual Técnico: descripción (no el código) de las clases implementadas, sus

atributos y sus métodos, así como el tipo de relación existente entre todas ellas. Se deben justificar las decisiones tomadas.

4. Manual de Usuario: descripción del uso de los programas realizados.
  5. Mejoras opcionales: descripción de las mejoras introducidas sobre la propuesta inicial, así como requisitos optativos.
  6. Observaciones: comentarios, problemas encontrados, críticas constructivas a la práctica 3.
- Se adjuntará un **disquete** o **cd-rom** que contenga los archivos **.java** creados por los alumnos **Si estos no compilasen o estuviese el disquete/cd corrupto, el grupo no obtendrá calificación en esta práctica. El código fuente debe compilar con la versión 1.5 de java (jdk 1.5)**

## 2.8 Evaluación de la práctica

La práctica se evaluará de acuerdo a los siguientes criterios:

- MEMORIA:
  - Presentación, claridad, ortografía: 0,1 puntos
  - Calidad manual técnico y usuario: 0,1 puntos
  - Calidad de justificaciones sobre implementación: 0,1 puntos
- FUNCIONALIDAD:
  - Implementación del comecocos: 0.3
  - Cargar y Salvar partida: 0.2
- CÓDIGO:
  - Calidad de la implementación: 0,1 puntos
  - Claridad del código y calidad de los comentarios: 0,1 puntos
- TEMAS OPCIONALES:
  - Incrementar ritmo de juego: 0,1 puntos
  - Mostrar vidas del protagonista: 0,15 puntos
  - Modificar la interfaz para mostrar la puntuación: 0,1 puntos

Mejoras adicionales, como la inclusión en esta práctica de actividades opcionales ya realizadas por el alumno en prácticas anteriores, serán tenidas en cuenta por los profesores de prácticas a la hora de evaluar.

Es requisito **fundamental** para calificar la práctica que el código entregado **compile sin errores** en jdk 1.5.