

JTartas

NORMAS DEL EXAMEN

1. Descargar el enunciado y los códigos suministrados.
2. Preparar el entorno del examen:
 - a. Consola.
 - b. Explorador de archivos.
 - c. Editor de texto.
 - d. Ventana del navegador con la doc de la API:
 - i. <https://docs.oracle.com/en/java/javase/16/docs/api/index.html>
3. Comenzar la grabación con OBS: **NO empezar el examen hasta no estar seguros de que la grabación se ha iniciado.**
4. Abrir el editor de texto.
 - a. Escribir la palabra POO.
 - b. Copiarla.
 - c. Pegarla dos veces.
5. Pulsar la combinación de teclas Windows + Tab.
6. Comenzar el examen.

Desarrollar un juego en el lenguaje de programación Java que siga la lógica que se describe a continuación.

El objetivo principal del juego es que no se caigan la tartas que caen del cielo gracias a la bandeja que controla el usuario de forma lateral gracias al teclado. EL bambú es el tercer tipo de objeto que existe en el juego que permite agrandar la bandeja recoge tartas.

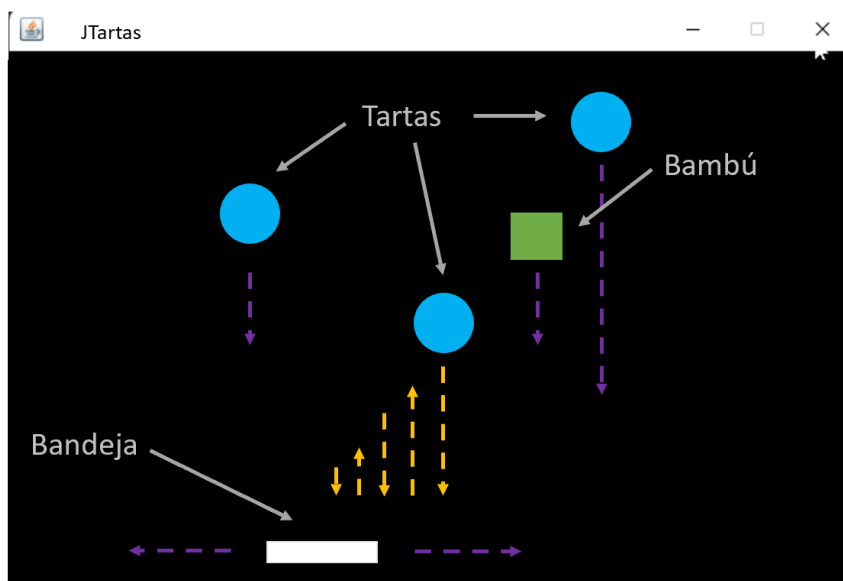


Fig 1. Ventana del juego y sus objetos

Descripción de los objetos del juego

Tartas:

- Solo pueden coexistir 3 objetos de este tipo en el juego.
- Caen del cielo desde una distancia aleatoria dentro del tercio superior de la ventana. Si el alto de la ventana son 600px, las tartas caerán en el rango [0, 200] pixels, por ejemplo.
- La posición respecto al eje X será aleatoria.
- Caerán del “cielo” a una velocidad constante prefijada al inicio de la creación del objeto entre [1, 4], siendo 1 la velocidad más lenta y 4 la más rápida. Flechas discontinuas moradas.
- Una vez que impacte con la bandeja rebotará (movimiento hacia arriba) hasta la altura inicial del objeto (posición Y en la que se creó) menos un desplazamiento de 20px. De esta forma simulará el bote de una pelota, donde cada rebote tendrá una altura menor que el anterior. Flechas discontinuas naranjas.
- En el caso de que no impacte con la bandeja caerá al suelo (desaparecerá), se generará una nueva tarta (siempre debe haber el mismo número de tartas en el juego) y se restarán 10 pixels al ancho de la bandeja.
- En el caso de que la tarta ya deje de botar por haber alcanzado el mínimo de altura, la tarta desaparecerá.
- Dimensión por defecto 30x30 pixels.

Bambú:

- Caerán del cielo desde una posición fija de altura (Y) y una posición aleatoria de ancho (X).
- Al impactar con la bandeja aumentará su ancho en 10 pixels.
- Si no se produce impacto, desaparecerá y se creará un nuevo objeto.
- La velocidad será siempre constante y fija para todos los objetos de este tipo.
- Dimensión por defecto 30x30 pixels.

Bandeja:

- El objeto que controla el usuario se mueve lateralmente gracias a las teclas A (izquierda) y D (derecha).
- Cuando la bandeja se reduzca hasta los 40 pixels de ancho finalizará el juego.
- Dimensión por defecto 20x80 pixels.

Aunque partamos de estos valores por defecto para esta solución y para todos los objetos de un mismo tipo, se deberá programar la solución para cualquier posibilidad de valores.



Fig 2. Estado del juego al llegar a un ancho de 40px

Diseño del juego

Además de la clase Generador, también se proporcionan la clase Juego y Tablero.

La clase Tablero posee un atributo llamado *fps* (frames por segundo) que nos permite calcular el tiempo que dejaremos pasar (delay) hasta volver a actualizar todos los personajes del juego con su nueva posición calculada.

Cálculo de impacto

La forma de saber si un objeto impacta con el personaje será mediante la intersección de los rectángulos que delimitan el área que ocupan los objetos. Una forma sencilla de resolver esta funcionalidad es gracias a la clase `java.awt.Rectangle` y su método *intersects*.

Herramientas que se podrán utilizar en el examen:

- Editor de texto.
- Compilador.
- JShell: si alguien desea ejecutar código Java de una forma rápida.
- No se podrá consultar ningún sitio web.
- Solo se podrá consultar la API que se encuentra en la siguiente URL:
<https://docs.oracle.com/en/java/javase/16/docs/api/index.html>
- No se podrá utilizar ningún IDE (Visual Studio Code, IntelliJ, Eclipse, etc.)

Se pide:

- Implementar todas las clases para dar respuesta al enunciado cumpliendo con los principios de orientación a objetos vistos en la asignatura.
- Seguir la estructura de paquetes que desee el alumno.
- Se podrá utilizar cualquier clase del JDK y estilo de programación visto en clase: funcional, por ejemplo.
- Se podrán utilizar import con `*`.
- El examen deberá ser grabado con OBS.