



Proyecto Intermedio

Para la entrega del proyecto intermedio de la asignatura los alumnos deberán entregar una primera versión del videojuego que realizarán para el trabajo final de la asignatura. Esta primera versión implementará los conceptos aprendidos en las prácticas realizadas hasta la fecha. Además, los alumnos deberán presentar a su clase de laboratorio el proyecto explicando brevemente las físicas y cómo se han cumplido los requisitos de la asignatura.

Como proyecto intermedio, para esta primera versión no es necesario presentar un videojuego completo.

Cada alumno tendrá 3 minutos para presentar su proyecto intermedio en la clase de laboratorio correspondiente a este proyecto y definida en el calendario de la asignatura. La presentación consistirá en un pequeño vídeo donde se muestre la ejecución del juego.

Requisitos

A parte de los requisitos básicos explicados en clase (implementación de la gravedad, entorno...) el proyecto entregado debe cumplir los siguientes requisitos. Estos requisitos deben ser fácilmente comprobables y ejecutables con el código proporcionado.

Requisitos de programación:

- A) Definición de una clase **partícula** con al menos los siguientes métodos y atributos:
 - 1) Uno de los métodos de integración vistos en clase.
 - 2) Masa como parámetro y al menos dos partículas definidas con distinta masa.
 - 3) Gestión de Fuerzas y aceleraciones teniendo en cuenta la masa.
- B) Implementación de la clase **proyectil** con un método de escalado de velocidad, energía y fuerzas como se vio en la práctica de proyectiles. Al menos debe haber implementado dos tipos de proyectiles.
- C) Definición de la clase **Sistema de Partículas** que use la clase partícula definida en A). Esta clase se encarga de gestionar conjuntos de partículas y los generadores (Fuerza y Partículas) de las mismas.
- D) Definición de los **Generadores de Partículas**.
 - 1) Los generadores de partículas usarán una partícula modelo que se define como atributo de los mismos. A partir de esta partícula modelo generarán nuevas partículas que serán copias de esta.
 - 2) La generación de partículas se realizará con una posición variable definida mediante al menos las dos distribuciones probabilísticas vistas en clase (Normal y Gaussiana).
 - 3) Cada clase Sistema de Partículas C) tendrá uno o varios Generadores de Partículas asignados y será esta clase la encargada de activarlos y desactivarlos.
- E) Definición de las clases de **Generador de Fuerza** de forma que se cumplan los siguientes requisitos.

- 1) El cálculo de fuerza se realiza de manera independiente para cada partícula. Es la partícula la encargada de llamar al Generador de Fuerza con su información.
- 2) Cada partícula puede recibir fuerzas de más de un Generador de Fuerzas.
- 3) Cada Sistema de Partículas C) llevará un registro de los Generadores de Fuerzas que estén activos para sus partículas (puede haber ciertos Sistemas que no queremos que les afecten ciertas fuerzas). Los Generadores de fuerza serán independientes de los sistemas de partículas (un generador de fuerzas puede afectar a varios sistemas).
- 4) Utilizar al menos dos tipos de generadores de fuerza. Uno de los generadores de fuerza (al menos) ha de implementar una fuerza que varíe con la posición de la partícula. Otra de las fuerzas, al menos, ha de variar con el tiempo (por ejemplo una explosión).

Requisitos mínimos de ejecución:

- Disparo de partículas (proyectiles) mediante una tecla asignada del teclado. Al menos un tipo de disparo se debe producir desde la cámara.
- Activación/Desactivación de los generadores de fuerzas mediante teclas. Cada Generador de Fuerza E) debe tener una tecla diferente asignada.

Entregables

Para este proyecto intermedio solo será necesario realizar la entrega del **código** de manera normal mediante git usando la etiqueta “Proyecto Intermedio”. A parte de esto se habilitará una entrega en el Campus Virtual donde cada alumno subirá el **vídeo** utilizado en la presentación y un **documento explicativo** del código. En este documento deberán aparecer instrucciones de cómo ejecutar el código (teclas asignadas a cada cosa y como jugar al juego) además de señalar en qué líneas/ficheros del código aparecen resueltos cada uno de los requisitos de programación. Así por ejemplo un documento explicativo podría ser:

Requisito A.1) → “particula.cpp” 120-160

Requisito A.2) → “particula.cpp” 178-179 atributo masa, “main.cpp” 57 y 58 definición de dos partículas con diferente masa.

Requisito C) → Definición de la clase en “SistemaParticulas.cpp”