

Tarea 3.A

EDGEY

Pablo Pizarro R.
ppizarror1@gmail.com

Introducción

Esta tarea consiste en diseñar y programar un juego similar (pero simplificado) al *EDGE Extended*, el cual tenía por objetivo recorrer un mapa con un cubo llegando desde un extremo a un fin recolectando unos *prismas*. Si el jugador llega al final del mapa sin recolectar todos los prismas repartidos por el mapa el jugador pierde.

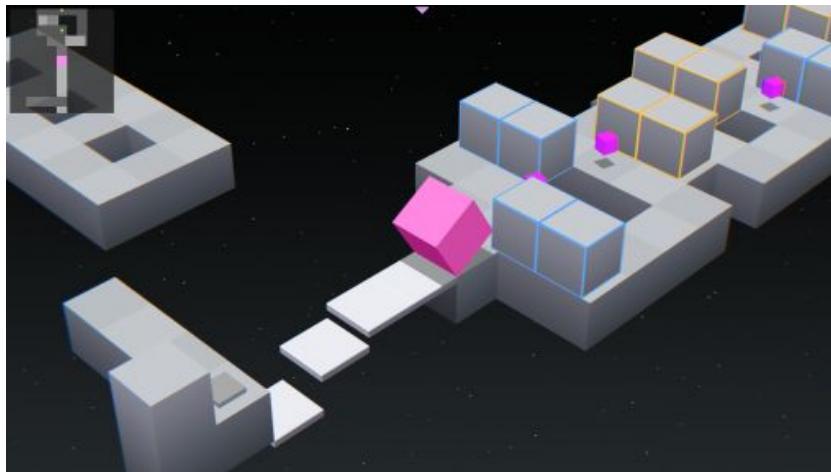


Figura 1: Ejemplo del EDGE: Un cubo (rosa) atraviesa un mundo recolectando prismas (cubos rosas pequeños).

Descripción del juego

El jugador puede seleccionar entre varios mapas distintos; este comienza en un punto fijo de cada mapa y debe recolectar distintos prismas repartidos por el mapa, una vez que obtiene todos los prismas puede llegar a una zona determinada (fin del mapa) para terminar la partida y ganar el nivel.

El mapa está lleno de peligros, el jugador puede siempre caer por el mapa (el jugador pierde), existen plataformas las cuales al moverse sobre ellas estas se caen (o se borran del mapa), muchos de los cubos se encuentran ocultos en el mapa y otras plataformas empujan al jugador en una determinada dirección (podrían empujarlo fuera del mapa).

Dibujar la escena

Se pide implementar un programa escrito en Pygame + PyOpenGL (implementación de OpenGL con Python e Pygame) el cual permita representar la escena 3D que define el juego. Esta escena debe considerar:

1. Un mundo hecho con cubos, este puede tener distintas alturas. NO pueden haber cubos flotantes. El mundo debe estar iluminado por una fuente puntual de luz. **(1.0pt)**
2. Un jugador (un cubo de color) el cual se puede mover en distintas direcciones, al moverse este debe rotar sobre una de sus aristas. **(1.0pt)**
3. Se deben dibujar distintos prismas de colores los cuales deben ser capturados por el jugador. **(0.5pt)**
4. Se deben dibujar plataformas que empujan al jugador y plataformas las cuales se caen al ser pisadas por el jugador. **(0.5pt)**

Implementación

1. El jugador se puede mover con las flechas del teclado (arriba, abajo, izquierda, derecha). El jugador se puede caer si se mueve fuera del mapa, el jugador además puede escalar entre los distintos niveles del mapa, sólo puede subir un cubo de altura. **(1.0pt)**
2. La cámara debe seguir al jugador, pero no puede rotar y permanece siempre a la misma altura. **(1.0pt)**
3. El jugador debe recolectar todos los prismas antes de llegar al final del mapa. **(0.5pt)**
4. Se deben programar las plataformas las cuales al ser pisadas estas se caen una vez son pisadas¹. **(0.3pt)**
5. Se deben programar las plataformas que empujarán al jugador en una dirección determinada. **(0.2pt)**

¹ Las plataformas se caerán sólo cuando el jugador las abandone.

Bonus

Además de los puntos anteriores usted puede hacer los siguientes bonus, **MAX 1pto**:

- Agregar música al juego, se debe considerar un tema de fondo, sonidos al moverse, al caerse del mapa, al caerse una plataforma o al ser empujado. **(0.5pt)**
- Añadir un menú que permita seleccionar entre los distintos mapas. **(0.5pt)**
- Añadir texturas a los cubos. **(0.5pt)**
- Utilizar `git` durante todo el desarrollo de la tarea. **(0.5pt)**

Informe

El informe debe incluir:

- Breve descripción del problema.
- Descripción de la solución.
- Resultados obtenidos
- Discusión sobre las dificultades encontradas.
- Conclusiones.

Formato PDF: Los informes entregados en otro formato serán evaluados con nota **1.0**

Condiciones de entrega

Fecha de entrega: Miércoles 28 de junio a las 23:59 hrs.

Ponderaciones: 70 % código, 30 % informe.

- La tarea es individual y las copias serán penalizadas.
- Entregas solo vía U-Cursos.
- Adjuntar archivo `Readme.txt` con instrucciones de ejecución.
- Recuerde adjuntar **TODOS** los archivos en cada entrega, ya que los auxiliares y ayudantes sólo tienen acceso a la última entrega de su tarea.
- **NO** se aceptarán archivos atrasados una vez el plazo haya finalizado, aunque esto implique que su tarea no ejecute.
- **NO** se aceptarán tareas sin informe.