

ACTA DE REUNIÓ No. 4

21 D'OCTUBRE DE 2021

ASSISTENTS: Sergi, Martí, Ricard, Hernan, Guillem, Alex

HORA D'INICI: 12:30

LLOC: Aula Q5/1005

1. Aprovació de l'acta anterior.

Aprovada, tots ho veiem correcte.

2. Discussió.

Començant la quarta sessió hem començat discutint sobre el material de la peça, mes que res per la lluminositat, amb Felipe, així que ho hem de tenir en compte per el tema de la lluento de cada bloc, nosaltres havíem pensat amb un Gouraund, però pot ser que les llums no les faci bé, per tant ho hem de tenir en compte la lluminositat de Phong e implementar-ho en un futur, si es cau.

Hem discutir si importarem la peça o nomes un bloc del color desitjat, hem decidit que serà millor importar un bloc del color i mitjançant opengl crear la peça desitjada.

A continuació hem parlat sobre les dimensions del taulell, no volem que sigui molt gran, per no tenir una experiència incomoda amb la càmera. Quedem en utilitzar 6x6x10 de moment, tot i fer un programa genèrica que es pugui adaptar a altres mides

Sobre l'skybox hem decidit canviar-ho perquè el color ens dificulta veure les peces. De totes maneres és un aspecte que deixarem per més endavant i de moment ens centrarem en aspectes més importants.

També discutim sobre la càmera. La posició de la càmera sobre l'eix Y no serà movable pel jugador. A masura que es va dibuixant el taulell, la posició de la càmera s'adaptarà a les posicions de les peces. Decidim fer-ho així per treure controls per part del jugador

i es pugui concentrar en moure les peces. La posició sobre l'eix Y serà sempre per sobre de la peça més elevada. Mentrestant, enfocarà al centre de l'últim nivell amb peces. Queda restant veure des de quina distància de l'eix X es veurà tot, doncs hem de poder veure bé el taulell i la peça que s'ha instanciat a dalt de tot.

S'ha decidit amb l'objectiu de facilitar la gestió de les peces que un cop col·locada la peça tots els blocs que la formen passin a formar part del taulell a la posició on queden. Per tant els tractarem com a apuntadors.

En quant al taulell, en comptes d'utilitzar una llista per guardar tots els blocs de les peces que es col·loquen, hem decidit fer una llista de llistes per tractar cada pis com una llista de blocs a emplenar. D'aquesta manera en comprovar si la llista conté un número de blocs igual al màxim de blocs d'un pis podríem esborrar el pis estalviant-nos les comprovacions en bucle si estiguéssim comprovant bloc a bloc.

3. Acords.

- Càmera: modificació lookat i posició en X e Y
- Codi:
 - funció generació de cada peça
 - moviment de la peça
 - main loop
 - creació d'una classe bloc a partir de la importació
 - funció eliminació de pis
- visualitzar mes d'un bloc en escena
- canviar tots els .obj perquè siguin blocs

Objectius per a la setmana següent:

DESENVOLUPAMENT	Descripció	Responsable	Participants	Durada (setmanes)	Grau Finalització
Càmera i espai	Creació d'una Camara base i un espai on assentar-la	Marti	Marti		0%
Subtasca: Adaptar-la	Adaptar l'entorn al que necessitem. Bloquejar eixos de moviment i angle de visió		Marti	2	30%
Subtasca: Distància de la camara	Distància des d'on veurem el joc i aplicar un zoom in i zoom out		Marti	2	70%
Fitxes	Disseny dels models de les peces	Ricard	Ricard Hernan	4	50%
Subtasca: Crear models de fitxes	Disseny de models en <i>blender</i>		Ricard	1	100%
Subtasca: Incorporar importador	Incorporar/testing d'un importador d'objectes obj		Ricard Hernan	1	100%
Instanciar fitxes a l'espai	Mostrar diferents fitxes importades a l'espai i que es vegin correctament		Ricard Hernan	2	50%
Passar de peça a bloc	Canviar els fitxers .obj		Ricard Hernan	1	0%
Texturització i entorn	Disseny de les textures de les peces	Guillem	Guillem Sergi	4	40%
Subtasca: Generar textures base	Dibuixar/crear Textures base per a les peces		Sergi	1	100%
Subtasca: Importar textures a <i>blender</i>	Aconseguir importar les textures a <i>blender</i> i entorn		Guillem Sergi	1	80%
Skybox	Modificar l'Skybox per posar un d'acord amb el tema del videojoc		Sergi, Guillem	1	100%

DESENVOLUPAMENT	Descripció	Responsable	Participants	Durada (setmanes)	Grau Finalització
Programació de la funcionalitat del projecte	Programar les mecàniques principals del joc	Alex	Alex	7	5%
Subtasca: Pensar l'estructures de Dades	Portar (i evaluar) Possibles estructures de dades que continguin el projecte		Alex	1	100%
Subtasca: Programar estructura de dades	Programar i implementar l'estructura de dades		Alex	1	0%
Subtasca: Funció genera una peça	Programar la funció que genera una peça		Sergi	1	0%
Subtasca: Moviment de la peça	Programar que una peça vagi caient		Guillem	1	0%
Subtasca: Main loop	Execució del programa		Alex	2	0%
Subtasca: Creació classe bloc a partir de la importació	Instancia el bloc en la classe		Sergi	1	0%
Subtasca: Eliminar pis	Eliminació d'un pis quant esta ple		Guillem	1	0%

Feina encara no assignada					
Texturització del entorn	Disseny de les textures secundaries	---			
Shaders	Programació de la il·luminació				
UI	Disseny de la interfície d'usuari				

Feina encara no assignada					
Testing Producte	Assegurar-se de que el producte funciona sota qualsevol circumstancia			2	

TESTING	Descripció	Responsable	Durda (setmanes)	Grau finalització

ÀNEX ESTRUCTURA DE DADES

Como guardar las piezas y el tablero

- **Bloques:** elemento principal que conforma el tablero y las piezas. De cada bloque, necesitamos guardar su posición, “si es movable”, tamaño (fijo) y si es candidato de recibir una colisión.
 1. Posición (Vector3/matrix)
 2. Movable (bool) ?
 3. Tamaño (cons int)
 4. Colisionable (bool)
- **Piezas:** guardamos las piezas como conjuntos de bloques. De cada pieza guardamos la forma y la estructura de bloques que la forman.
 1. Forma (char)
 2. Bloques (block[])
- **Tablero:** Se puede definir el tablero como un conjunto de bloques sin colisionador, se deberán definir los limitadores del tablero (costados).
 1. Bloques (Vector3/matrix)
 2. Limitadores[] (Vector3) ?
 3. Origen (Vector3)

Como definir la rotación

Utilizar un bloque como pivote de giro y definir un mapeado con la forma de la pieza tras cada rotación (rotación discreta y no animada).

Como comprobar las colisiones

Si se utiliza la variable “es movable”, se puede comparar la posición del bloque más bajo de la pieza que se está moviendo actualmente con la posición de los bloques “no movibles”, que harían de bloques ya colocados(límites). En el momento en que se detecta una colisión entre los diferentes tipos de bloque, se considerará el bloque actual como colocado y se modificará la variable de “es movable”.

Definir una variable que indique si la pieza es candidata de recibir una colisión, de esta manera reduciríamos la cantidad de bloques a comprobar.

Diseñar el bucle principal del juego

- Se genera una pieza de forma aleatoria (rotación incluida) en la posición origen del tablero. Para ello se puede hacer un mapeado de piezas identificadas con un número en concreto, después seleccionar una pieza a partir de una función de randomización de números, y por último, seleccionar su rotación de las diseñadas o calcularla.
- A cada segundo, se desplaza la pieza una posición hacia abajo, cualquier rotación de la pieza se hace al momento, por lo que se podrá mover la pieza varias veces por cada desplazamiento.
- Con cada desplazamiento se comprobará si se produce una colisión. Si no hay colisión de la pieza, se desplazará y se volverá al paso anterior. De lo contrario, la pieza no se desplazará y ésta pasará a formar parte de los límites de colisión.
- Se deberá comprobar si se ha rellenado algún piso del tablero, en caso afirmativo se deberá eliminar el piso y desplazar todos los bloques superiores una posición hacia abajo.

S'acaba la reunió a les 14:30.

Signatures:

Alex Castro Gastón

Sergi Bons Fuses

Guillem Centelles Pavon

Martí Caixal Joaniquet

Ricard Lopez Olivares

Hernán Capilla