



ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

TRABALHO PRÁTICO 1

Interação em Ambientes Virtuais | Semestre
Verão 21/22

Docente: Arnaldo Abrantes
Alunos: Miguel Silvestre nº45101 | Pedro Dias nº 45170

Trabalho efetuado no âmbito da cadeira letiva de IAV do Curso de Engenharia
Informática e Multimédia.

Índice

Índice de Figuras	2
1-Introdução	3
2-Desenvolvimento.....	4
2.1 – Construção do Mundo	4
2.1.1 – Texturas	4
2.1.2 – Ruído de Perlin.....	5
2.1.3 - Algoritmo de Flood Fill	6
2.1.4 – Parâmetros	6
2.1.4 – Desempenho.....	6
2.2 – Funcionalidades Adicionais.....	7
2.2.1 – Novos Blocos	7
2.2.2 - Skybox.....	7
2.2.3 - Música	8
3-Conclusão	9
4-Bibliografia	10

Índice de Figuras

Figura 1 - Atlas.....	5
Figura 2 - desempenho com diferentes parâmetros	6
Figura 3 - skybox pôr-sol	7
Figura 4 - skybox dia	7
Figura 5 - skybox noite	8
Figura 6 - skybox nascer sol.....	8

1-Introdução

Este trabalho visa a criação de um mundo num ambiente do tipo Minecraft em Unity.

Tendo como base o trabalho efetuado ao longo das aulas, onde se criou, com a utilização de técnicas procedimentais, mundos baseados em voxels, a geometria 3D do mundo (tris, quads, blocks, chunks), e se utilizou o ruído de Perlin para criar o terreno de forma processual, iremos construir a nossa própria versão acrescentando funcionalidades adicionais.

2-Desenvolvimento

2.1 – Construção do Mundo

De modo a ser possível termos um mundo infinito, é necessário este estar otimizado, de modo a diminuir a carga computacional. Assim, neste projeto, um mundo é criado continuamente em torno do agente à medida que este se movimenta e por conseguinte, parte do mundo é “destruído” se este se afastar consideravelmente.

Como dito anteriormente, este projeto visa a criação de um mundo Minecraft, ou seja um mundo baseado em cubos. Neste ambiente, existem faces que não são visíveis ao jogador, mas que ao serem criadas e renderizadas tornam a execução mais lenta. Assim, de forma a reduzir o número de triângulos, vértices, etc, e melhorarmos o desempenho, não processámos estas faces.

2.1.1 – Texturas

Para ser possível ao jogador distinguir os diferentes tipos de cubos, a cada um foi adicionado uma textura diferente, permitindo diferenciar a relva, a pedra, a terra, o ar, o diamante e o ouro. Isto foi realizado recorrendo a um atlas. Este, apesar de ser o mesmo utilizado nas aulas, tem texturas adicionais, nomeadamente diamante na posição (2,12) e ouro em (0, 13).



FIGURA 1 - ATLAS

2.1.2 – Ruído de Perlin

O ruído de Perlin é uma técnica muito usada em jogos e utiliza uma série de números parcialmente aleatórios com o objetivo de imitar objetos naturais, como o sol, nuvens, animações, terrenos, entre outros. Permite o controlo de elementos de pequena e grande escala e utilizamos este ruído de Perlin de forma a nivelar o nosso terreno. Quando maior for a sua regularização (smooth), mais nivelado fica o terreno.

2.1.3 - Algoritmo de Flood Fill

De modo a ser possível construir os diferentes tipos de blocos (chunks), utilizamos o algoritmo de Flood Fill que os desenha recursivamente. Em termos gerais, este algoritmo determina a área conectada a um nó dado em um vetor multi-dimensional de nós. Isto possibilita-nos a gerar mais do que um bloco do mesmo tipo, criando assim blocos vizinhos.

2.1.4 – Parâmetros

Para gerar o mundo foi tido em conta diferentes parâmetros. Os parâmetros foram calibrados da seguinte maneira:

- Os valores do Ruído de Perlin não foram alterados, pois foi considerado que estes mesmos valores já apresentavam resultados satisfatórios.
- Para fator *smooth* e o fator *octaves* foram criados arrays com diferentes valores para os mesmo e um dos valores é selecionado aleatoriamente permitindo assim termos mais diversidade no terreno, ou seja partes do terreno mais planas e partes do terreno com mais relevo.
- Foi usado um raio de 3 para geração do terreno

2.1.4 – Desempenho

Devido a aleatoriedade introduzidos é difícil de prever a variação de vértices e triângulos, no entanto foi feita uma estimativa correndo o programa uma vez. Os resultados foram os seguintes:

Raio	CPU(ms)	FPS	Vértices	Triângulos
1	3.5-4	265-300	20.1K-21.2k	10.6K-12.2k
2	3.6-4.1	180-260	28k-98.9k	14.5k-50.6k
3	3.9-4.3	165-246	54.7k-201.6k	27.9k-101.6k
4	4.3-5.4	144-220	161k-327.6k	82.2k-165.4k
5	4.6-5.8	130-211	351.1k-737.4k	177.2k-370.4k

FIGURA 2 - DESEMPENHO COM DIFERENTES PARÂMETROS

2.2 – Funcionalidades Adicionais

2.2.1 – Novos Blocos

Como dito anteriormente, acrescentamos dois tipos de blocos novos, diamante e ouro. Diamante pode ser encontrado em grutas misturado com a pedra. Ouro é criado através do utilizador. Este bloco pode ser criado em diversas posições (cima, baixo, frente, lado), consoante a tecla do teclado que o utilizador prime (E,R,T,Y,U,I). A posição do ouro criado depende da posição do utilizador.

2.2.2 - Skybox

Foi acrescentado neste trabalho uma skybox para tornar o ambiente do jogo mais apelativo e haver a existência de dia e noite no jogo. No total foram usadas 4 skyboxes que trocam entre si sequencialmente de 30 em 30 segundos para simular as 4 fases do dia (manhã, dia, pôr do sol, noite). O resultado final foi o seguinte:

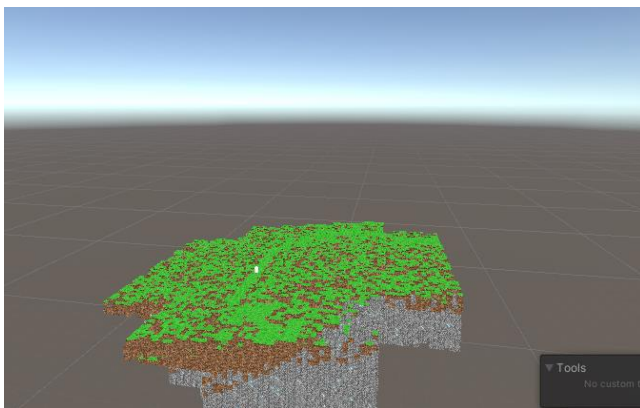


FIGURA 4 - SKYBOX DIA

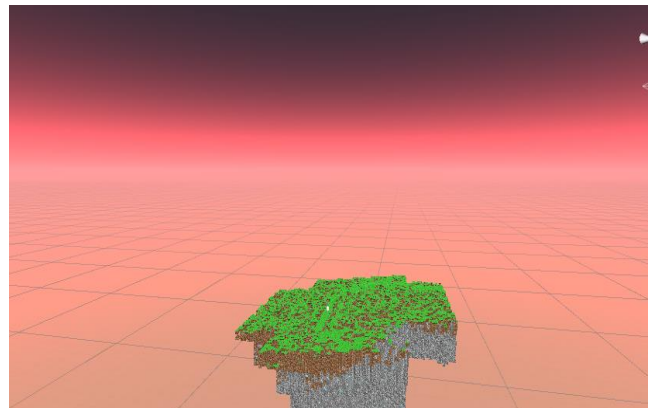


FIGURA 3 - SKYBOX PÔR-SOL

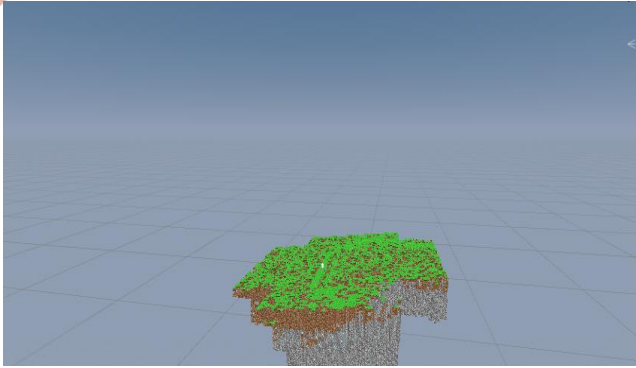


FIGURA 6 - SKYBOX NASCER SOL

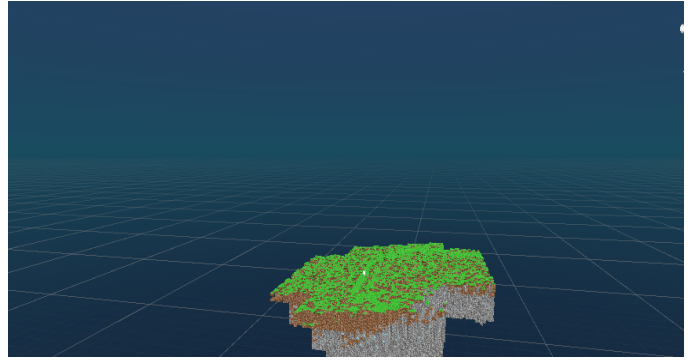


FIGURA 5 - SKYBOX NOITE

2.2.3 - Música

De forma ao jogo não ficar tão monótono e o jogador sentir um “vazio” enquanto está a jogar, adicionamos uma música ambiente durante a execução do jogo e um efeito sonoro na interação do utilizador, nomeadamente quando este cria um cubo de ouro.

3-Conclusão

Este trabalho foi interessante, pois toca num jogo bastante popular e com muito sucesso do nosso tempo atual.

A otimização é deveras importante, caso contrário não seria possível ao jogo fluir naturalmente, devido a carga computacional, congelando o ecrã por alguns segundos enquanto novas partes do mundo são criadas. Apesar do terreno ter sido criado com sucesso, este não está perfeito e existem melhorias a ser feitas, nomeadamente por vezes, alguns blocos ficam sobrepostos em cima uns dos outros e, por baixo das grutas um número pequeno de blocos não são construídos, sendo possível ao jogador cair e não conseguir proceder o jogo.

Neste tipo de jogos, existe um número incontável de funcionalidades adicionais que podem ser acrescentadas, o limite é a nossa imaginação. No entanto, salienta-se a possibilidade de o jogador conseguir destruir vários tipos de blocos. Infelizmente, o tempo não foi suficiente para conseguirmos realizar esta interação.

Apesar da existência de alguns problemas e haver espaço para melhoramento, consideramos que o trabalho cumpre os requisitos e foi realizado com sucesso.

4-Bibliografia

1. Vídeos das aulas no Youtube fornecidos pelo docente - Arnaldo Abrantes.
2. https://en.wikipedia.org/wiki/Perlin_noise
3. https://en.wikipedia.org/wiki/Flood_fill