

A Draughts Game based on WebGL

Martim Neves 88904

Tiago Dias 88896

Resumo – O presente artigo apresenta o desenvolvimento de uma aplicação 3D intuitiva que permita jogar ao jogo das damas utilizando a API JavaScript WebGL.

O artigo começa por estabelecer as principais regras do jogo das Damas. De seguida, são apresentados os principais requisitos da aplicação 3D considerados ao longo do desenvolvimento do projecto e como foi feita a sua implementação. Finalmente, descrevem-se os resultados obtidos.

I. INTRODUÇÃO [1]

O jogo de damas é disputado entre dois adversários que movem peças (peões ou damas) num tabuleiro quadrado denominado “tabuleiro de damas”.

A. Tabuleiro das Damas

O “tabuleiro de damas” é composto por uma rede de 8 colunas horizontais e 8 colunas verticais que formam 64 quadrados iguais, alternadamente claros (preferencialmente brancos ou castanhos claros) e escuros (preferencialmente pretos ou castanhos escuros).

O tabuleiro é colocado entre os jogadores de tal forma que o quadrado do canto inferior direito de cada jogador seja escuro.

B. Jogo

No início do jogo, um dos jogadores dispõe de 12 peões brancos (ou castanho claro) e o outro de 12 peões negros (ou castanho escuro). O jogador com as peças brancas inicia o jogo.

C. Tipos de Peças

Existem 2 tipos de peças: peões e damas. Os peões são peças normais e cada jogador começa com 12. Sempre que um peão atinge uma das quatro casas do extremo oposto do tabuleiro, o jogador da cor contrária colocará um dos peões já capturados sobre ele, coroando-o como dama.

D. Movimentos das peças

As peças apresentam dois tipos de movimento: sem captura e com captura:

- Sem captura – Os peões movem-se sobre o tabuleiro, sempre para a frente e numa das suas diagonais, ocupando uma das casas contíguas vagas, não tomando no seu percurso qualquer peça de cor contrária. As damas, por sua vez, conseguem executar este movimento não apenas para a frente, mas também para trás e a casa para a qual se deslocam não tem de ser contígua àquela na qual se encontram.
- Com captura – Sempre que uma peça, ao iniciar o seu movimento tenha, numa casa contígua, no caso dos peões, ou não, no caso das damas, uma peça de cor contrária, existindo a seguir casa vaga que possa ocupar, saltará por cima da peça e capturará a mesma; se, ao pousar nessa casa, existir nova peça seguida de casa que possa ocupar (nos mesmos moldes nos quais é efetuada a primeira captura), continuará o seu trajeto, capturando as peças que se encontrem nessa situação.

E. Regras de captura

- Lei da Obrigatoriedade - A captura é obrigatória;
- Lei da Quantidade - Em hipóteses simultâneas de captura, é obrigatório tomar o maior número de peças possível;
- Lei da Qualidade - Em hipóteses simultâneas de captura de “um mesmo número de peças” é obrigatório tomar as peças com maior valor, sendo o valor da dama superior ao do peão.

F. Vitória

A «vitória» ocorre quando:

- O jogador captura todas as peças do adversário;
- O adversário fica com as peças “prisoneiras” na vez de jogar não podendo efetuar lance;
- O adversário declara o abandono.

II. REQUISITOS

A. Requisitos de Jogo

Os requisitos de jogo apresentados na Introdução foram cumpridos e implementados, em específico:

- Visualização do tabuleiro de jogo e de peças;
- Selecção de peças e posições para movimentação das peças;
- Validação da movimentação de uma peça:
 - Um jogador apenas pode mover as suas próprias peças;
 - Essas movimentações só podem ser feitas de acordo com as regras do jogo;
- Implementação da movimentação da captura;
- Sistema de pontuação para uma mais fácil visualização de quem está em vantagem;

Algumas simplificações foram consideradas de modo a conseguir implementar todo o projeto em tempo útil:

- As peças não podem transformar-se em damas;
- As capturas não cumprem as Leis (ver Secção 1E);
- Uma vitória apenas pode ocorrer quando um jogador captura todas as peças do adversário.

B. Requisitos da aplicação 3D

O objectivo do projecto é o desenvolvimento de uma aplicação **3D interactiva** para jogar o jogo das Damas. Isto implica a consideração de um conjunto de requisitos adicionais:

- A visualização do tabuleiro de jogo e das peças tem de ser em 3D;
- Funções de ampliação/redução, rotação e translação devem poder ser aplicadas ao tabuleiro de jogo e peças;
- Visualização de 2 pilhas com as peças capturadas por cada jogador;
- Deve ser possível mudar a cor inicial das peças de cada equipa;

III. IMPLEMENTAÇÃO

A. Modularização

Para poder ter um projeto bem estruturado e modular foram criados os ficheiros presentes na tabela 1.

Para executar o jogo implementado, é necessário abrir o ficheiro `indx.html`.

Ficheiro	Propósito
<i>index.html</i>	Página <i>HTML</i> onde o JavaScript é executado e através da qual é realizada toda a interação entre o sistema e o utilizador
Board.js	Definição das classes <code>Board</code> , <code>Slot</code> e <code>Draught</code> (propriedades e métodos)
<i>DrawModel.js</i>	Função <code>drawModel()</code>
<i>DrawScene.js</i>	Função <code>drawScene()</code>
<i>Illumination.js</i>	Definição da iluminação do canvas e função <code>drawModelIllumination()</code> – não utilizado
<i>InitBuffers.js</i>	Função <code>initBuffers()</code> e funções respetivas para cada classe definida no ficheiro <code>Board.js</code>
<i>Main.js</i>	Controlo da execução do JavaScript
<i>Utils/animation.js</i>	Funções <code>animate()</code> e <code>tick()</code>
<i>Utils/eventHandlers.js</i>	Funções para a gestão dos eventos de teclado e rato
<i>Utils/fps.js</i>	Funções auxiliares para cálculo de <i>fps</i> da simulação
<i>Utils/initShaders.js</i>	Funções para criação e <i>linking</i> dos <i>shaders</i>
<i>Utils/math.js</i>	Funções matemáticas auxiliares
<i>Utils/webgl-utils.js</i>	Funções auxiliares para compatibilidade em <i>Browsers</i>
Utils/css/*	Estilos da página <i>HTML</i>
<i>Utils/icons</i>	Pasta com as imagens utilizadas na página web

Tabela 1 – Ficheiros do projeto

B. Implementação da Lógica de Jogo

A lógica do jogo das Damas está implementada no módulo *Board*. Nesse módulo estão definidas as 3 principais entidades, o *Board*, que contém por sua vez *Slots* (posições do tabuleiro) e *Draughts* (peças).

Estas entidades contêm as informações e a lógica necessária para a sua visualização, selecção (no caso dos slots) e movimentação (no caso das peças).

Visualização do tabuleiro de jogo e das peças

A visualização de um tabuleiro de jogo e das peças (Fig. 1) é feita utilizando a informação de vértices, índices dos vértices de cada triângulo e informações de cor armazenadas nas 3 entidades.

Durante o jogo, algumas dessas propriedades, nomeadamente a cor, podem ser alteradas (por selecção de posição nos slots ou por alteração das definições, no caso das peças).

As peças do tabuleiro são desenhadas com recurso a fórmulas matemáticas, de modo a criar todos os vértices necessários nas posições necessárias para que o resultado final seja um cilindro.

Seleccção de peças e posições

A selecção de uma peça e de posições é feita utilizando o teclado. O rato foi considerado como alternativa, mas tornava inviável jogar com o tabuleiro em posições distintas da posição por omissão. Além disso, o rato já é utilizado para rodar o tabuleiro e as peças, pelo que foi optado pelo uso das setas para a movimentação das peças.

Para seleccionar uma peça é necessário, utilizando as teclas direccionais (ie. Setas), seleccionar a sua posição. (Fig. 4)

De seguida, é necessário pressionar a tecla Enter, ficando a peça seleccionada (Fig. 5).

Após isso basta seleccionar a nova posição e efetuar a movimentação pressionando a tecla Enter novamente (Fig. 6). De notar que movimentações inválidas são ignoradas.

Se ao movimento do peão corresponder uma captura, a peça capturada desaparece e é colocada de lado, na pilha correspondente (Fig. 7).

Validação da movimentação de uma peça

Como descrito na secção 1D – Movimentos das peças, as peças do jogo de damas têm movimentos muito bem definidos e que podem ser divididos em duas categorias:

movimento sem captura (avanço) ou movimento com captura (captura). A lógica complica-se um pouco ao considerar o alcance acrescido de uma dama. O mesmo se aplica caso tivessem sido aplicadas as leis relativas às capturas de peças adversárias (ver *Secção 1E*). No entanto, como a lógica de jogo foi aplicada meramente a peões, a validação de um movimento tornou-se relativamente simples.

Após analisar a qual dos jogadores cabe a jogada, ficando assim definida a direcção para a qual a peça é suposto avançar, verifica-se se o avanço (no eixo XX) foi de uma ou duas unidades. No primeiro caso, considera-se que a jogada só pode tratar-se de um avanço (sem captura) enquanto que no segundo considera-se o cenário de captura de peça. São analisadas as casas de início e fim de movimento da peça seleccionada e, no caso de cenário de captura, a peça que se encontra entre as casas analisadas é retirada do tabuleiro, pois foi capturada, e é adicionada à pilha de peças capturadas.

Implementação da movimentação de captura

Após a criação de estruturas de dados e designação de um espaço do tabuleiro dedicado, as peças capturadas passaram a sofrer, após captura, uma translação para o topo de uma pilha que se encontra do lado direito da margem do tabuleiro correspondente ao autor da jogada. Esta pilha vai crescendo de cada vez que se verifica uma nova captura (Fig. 7 e 8).

Manutenção de um sistema de pontuações ao longo do jogo

No seguimento do anteriormente analisado movimento de captura, foi criado um sistema de pontuações para auxiliar os jogadores a rapidamente avaliarem o estado do jogo sabendo instantaneamente quantas peças foram já capturadas de parte a parte. Este sistema permite ainda o término do jogo pois, de acordo com o referido na *Secção 1F e 2A*, o jogo termina aquando da captura de todas as peças de um dos jogadores.

C. Implementação da aplicação 3D

Funções de ampliação/redução, rotação e translação

Funções de ampliação/redução, rotação (nos eixos do XX, YY, ZZ) e de translação (nos eixos do XX, YY e ZZ) foram implementadas e aplicam-se à globalidade do tabuleiro e podem ser controladas através da interface Fig. 10 – Opções e informações de jogo (ou através do rato e clicando no botão direito do rato).

Isso significa que foram aplicadas transformações à matriz de transformação global através de multiplicações sucessivas de matrizes correspondentes às diferentes funções e parametrizadas com os valores das variáveis globais presentes no módulo *Board.js*.

Transformações locais, ie. aplicadas só a parte dos modelos (Board, Slots ou Draughts/peças) foram consideradas, mas apenas na situação de movimentação das peças, no qual se aplicariam translações. Contudo, e tendo em conta a forma como os vértices de cada peça foram gerados, essa abordagem foi impossível de implementar em tempo útil.

Visualização de 2 pilhas com as peças capturadas

Para que houvesse *feedback* visual relativamente às peças capturadas por ambas as equipas, foram criadas, como referido anteriormente, estruturas de dados que replicavam fielmente o empilhar das peças capturadas que se vê em jogos reais. Como a maioria das pessoas é destra, as pilhas de peças capturadas costumam ficar do lado direito do tabuleiro (lado da mão dominante). Assim, as peças de uma equipa que forem capturadas sofrem uma translação para a pilha de peças capturadas pela equipa adversária (Fig. 7 e 8).

Alteração do aspecto das peças

A qualquer momento, é possível alterar o aspeto das peças de uma equipa, utilizando a interface disponível (Fig. 10).

Inicialmente foi considerado a possibilidade de mudar o aspeto através da alteração da textura ou iluminação aplicados, mas concluiu-se que, dado o elevado número de vértices de cada peça, seria mais vantajosa a aplicação de conjuntos de 3 cores diferentes, criando um efeito de gradiente de cores.

2 cores estão disponíveis para cada equipa, vermelho e cinzento para as peças da equipa 1, preto e verde para as peças da equipa 2 (Fig. 1 e Fig. 2).

Na secção 1B foi dito que o jogador com as peças brancas inicia o jogo. No entanto, escolhemos não dar peças brancas a nenhum jogador, para haver uma melhor perceção dos diferentes elementos. Em vez disso, quem começa o jogo é o jogador 1, cujas peças são vermelhas, inicialmente.

É possível adicionar mais cores através da alteração do método `setColor` da entidade `Draught`.

IV. RESULTADOS

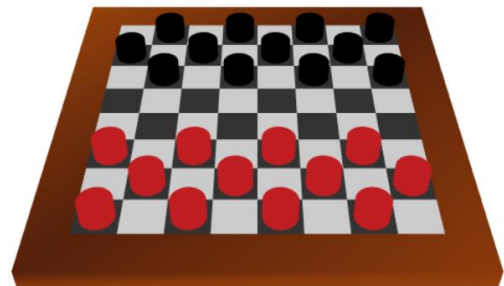


Fig. 1 – Início de jogo

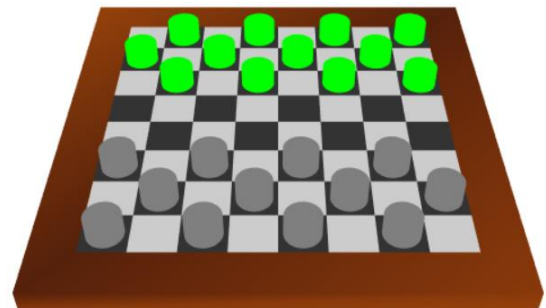


Fig. 2 – Início de jogo com aspeto alternativo de peças

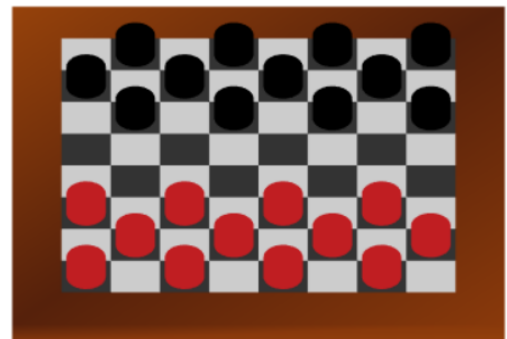


Fig. 3 – Início de jogo com o tabuleiro na perspetiva ortogonal

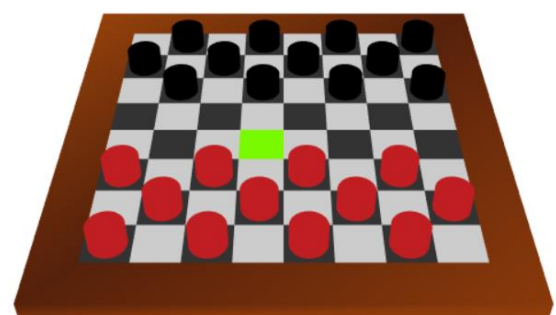


Fig. 4 – Seleção de uma posição

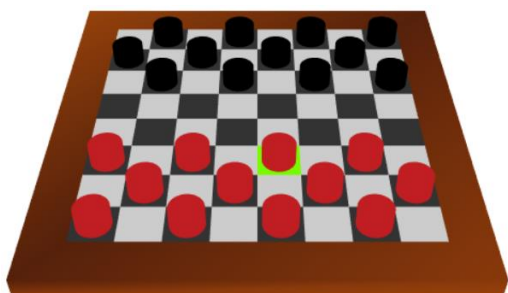


Fig. 5 – Seleção de uma peça

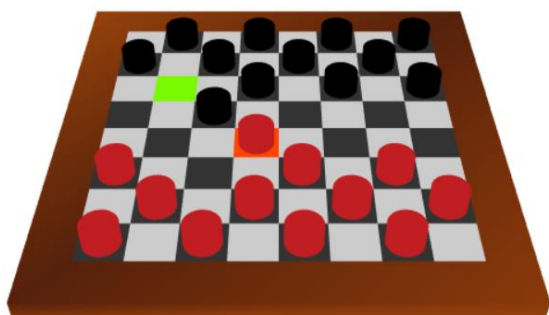


Fig. 6 – Seleção de uma posição (antes da captura)

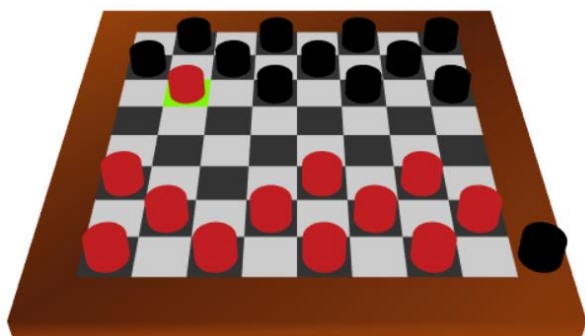


Fig. 7 – Após a captura, agora com uma pilha de peças capturadas

Esta página diz

Game Over! Congratulations to Team 2

OK

Fig. 8 – Fim de jogo

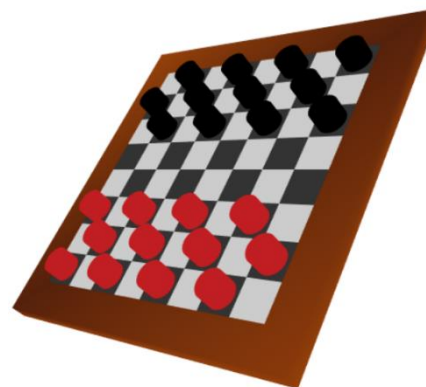


Fig. 9 – Tabuleiro rodado (exemplo)

Game Info ?

Turn : Team 1

Score : 0 - 0

FPS : 57

Restart Game

Restart View

Move



Scale



Rendering Mode

Filled Triangles

Projection type

Perspective Projection

Draughts Colors

Team 1:

Red

Team 2:

Black

Rotate

XX

On / Off

Change Direction

Slower

Faster

YY

On / Off

Change Direction

Slower

Faster

ZZ

On / Off

Change Direction

Slower

Faster

Fig. 10 – Opções e informações de jogo

REFERÊNCIAS

- [1] Federação Portuguesa de Damas, Regras do Jogo das Damas Clássicas, <http://www.fpdamas.pt/Downloads/REGRAS%20DAS%20DAMAS%20CL%C3%81SSICAS.pdf>, 2013.
- [2] Material fornecido no âmbito da Unidade Curricular de Computação Visual, lecionada pelos professores Joaquim Madeira e Paulo Dias, 2020