

FACULDADE DE ENGENHARIA
DA UNIVERSIDADE DO PORTO



MANUAL DE UTILIZADOR

LABORATÓRIO DE APLICAÇÕES COM INTERFACE GRÁFICA

MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA E
COMPUTAÇÃO

LYNGK

Autores:

Afonso Jorge Ramos

Miguel Sozinho Ramalho

up201506239@fe.up.pt

up201403027@fe.up.pt

Dezembro 2017

Resumo

O presente Manual de Utilizador descreve a abordagem utilizada na replicação do jogo de tabuleiro LYNGK numa interface gráfica após a sua implementação na linguagem de programação em lógica PROLOG, desenvolvida na cadeira de PLOG.

Por fim, considera-se que este projeto serviu como rampa de lançamento das nossas capacidades de implementação de aplicações com interface gráfica, assim como, de abordagem de problemas lógicos na implementação de um jogo com todas as funcionalidades requeridas, o que também nos despertou o interesse de continuar a desenvolver aplicações com interface gráfica e formas de usar esta ferramenta na implementação das mesmas.

Conteúdo

1	Introdução	3
2	Lyngk	4
2.1	História	4
2.2	Regras	4
2.2.1	Preparação	4
2.2.2	Peças, Reivindicação de Peças & Cores	4
2.2.3	Movimentos	5
2.2.4	A Regra Lyngk	6
3	Utilização	8
3.1	Execução	8
3.2	Instruções	8
3.2.1	Ecrã Inicial	8
3.2.2	Definições e Algumas funcionalidades	8
3.2.3	Ecrã de Jogo	9
3.2.4	Mensagens do Servidor	10

1 Introdução

No âmbito da unidade curricular de Laboratório de Aplicações com Interface Gráfica, do curso Mestrado Integrado em Engenharia Informática e Computação, foi-nos sugerido o desenvolvimento de uma interface utilizando a biblioteca WebCGF em *javascript* para um jogo desenvolvido usando a linguagem de programação PROLOG.

Dos dois elementos do grupo tínhamos duas opções disponíveis de jogos desenvolvidos em Programação em Lógica, o jogo Lyngk e o jogo Corrida de Reis, nesse sentido o grupo escolheu o jogo Lyngk, por ser uma opção bastante mais desafiante face à sua implementação e por posar um jogo mais fora do comum, visto que o Corrida de Reis utilizaria um tabuleiro de xadrez. Não obstante, esta interface mostrou ser um bom desafio, levando-nos a conhecer melhor os conceitos subjacentes à implementação de interfaces gráficas.

O objetivo deste trabalho foi a aplicação dos primeiros conceitos interiorizados nas aulas teóricas e desenvolvidos nas aulas práticas da cadeira. Este método de avaliação torna-se importante pois permite-nos avaliar os conhecimentos que adquirimos até então e saber se somos ou não capazes de aplicar os conceitos teóricos.

2 Lyngk

2.1 História

Kris Burm (Antuerpia, 1957) é um inventor de jogos Belga, especializado em jogos de tabuleiro abstratos. É conhecido pela série de Jogos GIPF, um dos quais é o Lyngk. O Lyngk é, alias, o último dos jogos desta série (até agora), e é tido como um agregar das ideias expostas nos outros jogos. Foi lançado em 2017, mas rapidamente se tornou famoso, dada a reputação do seu autor e da série de jogos que integra.

2.2 Regras

2.2.1 Preparação

O jogo deve ser jogado por duas pessoas. No princípio, o tabuleiro deve ser preenchido com 8 peças de cada cor (azul, marfim, vermelho, verde e preto) mais 3 peças *wild*, dispostas aleatoriamente, como mostra a Figura 1.

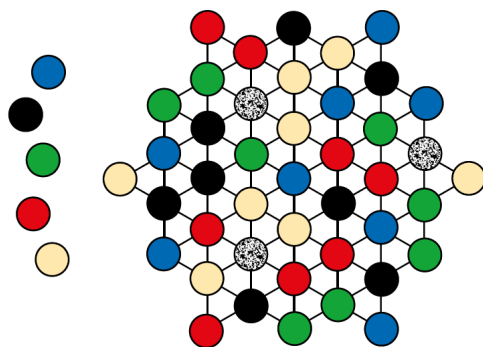


Figura 1: Exemplo de um tabuleiro no estado inicial

5 peças (uma de cada cor) devem ser deixadas de parte, junto ao tabuleiro. Seguidamente, deve ser sorteado o primeiro jogador.

2.2.2 Peças, Reivindicação de Peças & Cores

- As três peças *wild* devem ser consideradas *jokers*, isto é, elas têm a capacidade de representar qualquer peça do tabuleiro. Estas peças são passivas. Não podem

ser utilizadas para jogar, e só podem ser movidas como parte de uma pilha.

- No início do jogo, todas as peças no tabuleiro são neutras, e não pertencem a qualquer dos jogadores. Isto significa que ambos os jogadores as podem usar.
- Durante o decorrer do jogo, cada jogador pode reclamar até duas peças. Assim que uma peça é reclamada por um jogador, deixa de ser neutra. Desse momento em diante, só o jogador que reclamou a peça pode jogar com peças dessa cor.

Nota: as peças ao lado do tabuleiro servem para ser reclamadas, e não podem entrar no tabuleiro.

- Os jogadores podem reclamar uma cor em qualquer altura do jogo, mas só durante o seu turno, e antes de fazerem a sua jogada. Para reclamar uma cor, o jogador deve retirar uma peça do lado do tabuleiro para junto de si.
- Um jogador só pode reclamar uma cor de cada vez, isto é, não pode reclamar duas cores no mesmo turno.
- Quando ambos os jogadores tiverem reclamado duas cores, a última cor permanece neutra. Ambos os jogadores podem continuar a jogar com peças dessa cor até ao final do jogo.

2.2.3 Movimentos

1. Os jogadores tomam turnos. A cada jogada, o jogador deve mover uma peça ou uma pilha de peças. Para fazer um movimento, podem escolher qualquer peça ou pilha de cor neutra, ou de uma das cores que tenham reclamado.
2. As pilhas de peças devem ser sempre movidas atomicamente. A peça no topo da pilha determina a cor da pilha.
3. Um movimento, seja com uma cor neutra, ou com uma cor reclamada, deve terminar sempre num espaço ocupado, ou seja, em cima de outra peça ou pilha. Um movimento pode terminar em cima de uma peça/pilha adjacente ou em cima de uma peça/pilha que possa ser alcançada jogando em linha reta, cruzando só espaços vazios, isto é, não é permitido saltar por cima de outras peças/pilhas.
4. Há uma outra forma de fazer movimentos, descrita mais abaixo, na secção Regra Lyngk.
5. Uma pilha pode ter até 5 peças. A regra-chave é que uma pilha tem de ter todas as peças de cores diferentes: nunca pode haver duas ou mais peças da mesma cor numa pilha. No entanto, podem haver até 3 peças brancas numa pilha, dado que estas tomam as cores de peças que não estejam já na pilha.

6. Uma única peça neutra só pode ser movida para cima de uma outra peça de qualquer outra cor. Por outras palavras, pode ser movida para cima de uma peça branca, para cima de outra peça neutra, ou para cima de uma peça que tenha sido reclamada por qualquer um dos jogadores. Uma única peça neutra não pode saltar para cima de uma pilha.
7. Uma pilha com uma peça neutra no topo pode saltar para cima de qualquer peça, ou pilha cuja altura seja igual ou inferior à sua.
8. Uma única peça de uma cor reclamada, ou uma pilha de uma cor reclamada podem ser movidos para cima de qualquer outra peça/pilha, desde que a pilha resultante não fique com um número de peças superior a 5, e desde que todas as cores nessa pilha sejam diferentes.
9. Quando um jogador completa uma pilha de 5 peças, e a peça no topo é de uma cor que esse jogador tenha reclamado, esse jogador deve retirar essa pilha do tabuleiro e colocá-la junto a si, de modo sempre visível ao oponente. Cada pilha removida vale 1 ponto no final do jogo.
10. Quando um jogador completa uma pilha de 5 peças, com uma cor neutra em cima da pilha, a pilha passa a comportar-se como um obstáculo, e não sai de cima do tabuleiro. Esta pilha não vale como ponto para nenhum dos jogadores.
11. Um jogador não pode passar a jogada, a menos que não tenha movimentos possíveis.
12. Se um jogador estiver impossibilitado de jogar, o outro jogador deve continuar a jogar até que este, também, fique sem opções de jogo. Caso um jogador, que anteriormente tenha passado, tenha a possibilidade de jogar, então deve fazê-lo.

2.2.4 A Regra Lyngk

Na secção abaixo, tanto peças como pilhas são designadas como "peças", dado que a regra Lyngk é válida para ambas. A Figura 2 mostra um diagrama da aplicação da regra Lyngk, durante a jogada do Jogador B.

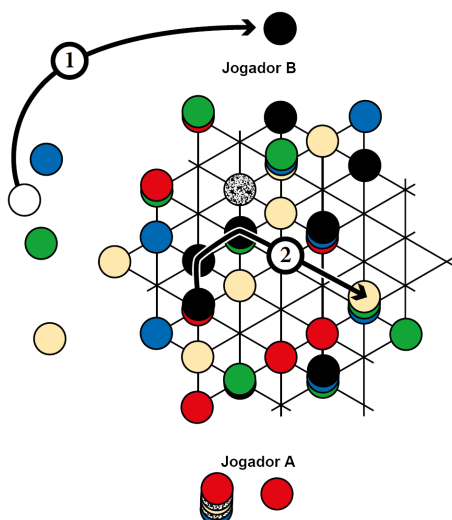


Figura 2: Exemplo da Regra Lyngk

O Jogador A já tem uma pilha, estando em vantagem, contudo o jogador B tem a possibilidade de igualar o seu adversário através da regra Lyngk. Primeiro, deve reclamar a cor preta (1), e depois faz um movimento triplo (2) e completa uma pilha de 5 peças. Para fazer este movimento múltiplo, todas as peças-Lyngk têm de ser da mesma cor, e esta deve ser uma das cores reclamadas.

1. A regra Lyngk só pode ser utilizada quando o jogador estiver a mover uma peça reclamada.
2. A regra diz que as peças de uma dada cor estão ligadas (Lyngk), mas só se estiverem posicionadas de tal modo que seja possível juntá-las com um movimento normal. Um jogador pode usar peças dessa cor reclamada para fazer um movimento duplo, ou até um movimento triplo ou quádruplo, usando-as como ligações para outras peças no tabuleiro.
3. A regra aplica-se do seguinte modo: um jogador pode mover uma peça de uma cor reclamada até uma outra peça da mesma cor, onde não a empilha, antes a usa como um ponto-Lyngk, o que significa que deve fazer um segundo movimento a partir daí. Assim, da peça alcançada, o jogador deve mover (a peça original) para uma peça adjacente, ou para uma peça acessível em linha reta. A peça jogada deve ser empilhada na peça de destino. No entanto, se esta segunda peça for da mesma cor da peça jogada originalmente, o jogador deverá fazer um terceiro movimento a partir daí, e assim por diante, até que a peça chegue a um sítio onde possa aterrar. (Ver Figura 2)

3 Utilização

3.1 Execução

1. Abrir SICStus, consultar o ficheiro de PROLOG *server.pl* e chamar o predicado *server.*;
2. Criar servidor local utilizando o ficheiro executável do Mongoose e abrir a pasta LYNGK;

3.2 Instruções

3.2.1 Ecrã Inicial

Depois de completar as instruções de execução, somos recebidos pelo seguinte ecrã.

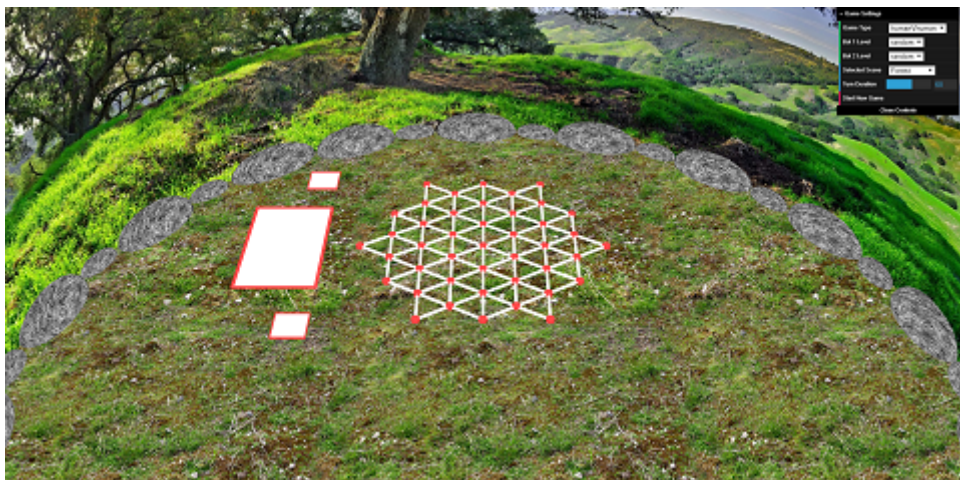


Figura 3: Ecrã inicial.

Aqui, podemos desde já observar toda a geometria presente na cena, excluindo os elementos de jogo, como por exemplo, as peças que só serão criadas com o início do jogo, visto que as mesmas são geradas aleatoriamente.

3.2.2 Definições e Algumas funcionalidades

Para qualquer mudança nas definições podemos aceder à interface presente no canto superior direito, tal como podemos observar na figura 4 que possui opções como, tipo

de jogo, *ie*, humano contra humano, bot contra humano, ou ainda, bot contra bot; Nível de Inteligência de ambos os bots, no caso de se tratar de um jogo entre dois bots, ou apenas do primeiro, caso seja bot contra humano; Cena selecionada, que pode variar entre as duas criadas para o jogo, no qual uma delas possui rotação da câmara, sendo, por isso orientada a jogos entre dois jogadores, enquanto que a outra se fica por uma câmara presa, sendo por isso orientada para jogos com bots; e ainda, duração do turno, permitindo a alteração do tempo de turno para ajustar o tempo de jogo como o utilizador pretender, mas não permitindo, como é óbvio, a alteração do mesmo a meio do jogo.

Possuímos também a funcionalidade do **undo** e do **filme de jogo**, sendo que o último é acedido através da interface abaixo, enquanto que o undo é controlado através da utilização do comando de teclado *CTRL+Z*.

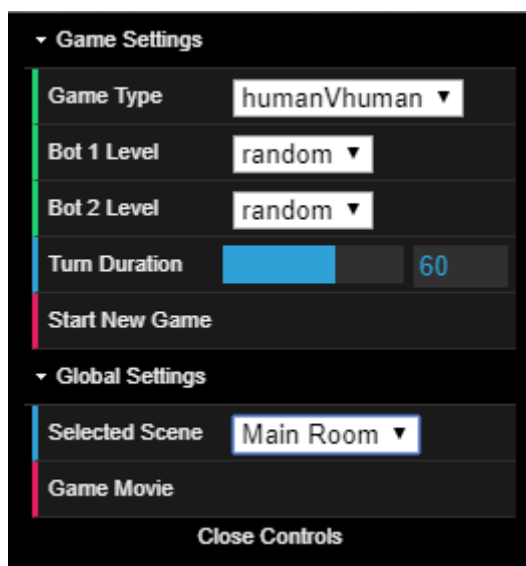


Figura 4: Definições de Jogo disponíveis.

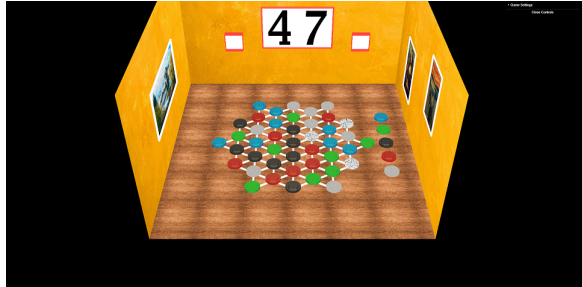
3.2.3 Ecrã de Jogo

Depois de iniciarmos um jogo, todas as peças são carregadas, sendo, portanto todas colocadas em cima do tabuleiro. Olhando, por exemplo, para o tabuleiro da floresta, podemos observar que à direita do tabuleiro podemos ver as cores disponíveis para seleção por parte do jogador, sendo que cada jogador só pode selecionar uma por jogada, já à esquerda vemos os marcadores, assim como o temporizador. O valor das pontuações que vão aparecendo nos marcadores são calculados em função do número e

altura das pilhas que cada jogador tem, ou seja, daquelas cuja cor de cima corresponde à do jogador, daí a pontuação de ambos jogadores no início do jogo ser 20.



(a) Cena da Floresta.



(b) Cena da Sala.

Figura 5: Exemplo das cenas usadas após o início de jogo.

Para efetivamente jogar o jogo, basta selecionar a peça que pretende mover, pois toda ele é selecionável, o que é demonstrado pelo shader amarelo a ela aplicada, onde depois podemos observar as várias jogadas possíveis, demonstrado pelo shader preto a elas aplicado.

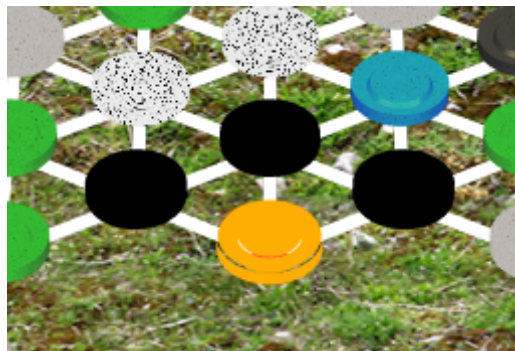


Figura 6: Seleção da peça e jogadas possíveis.

3.2.4 Mensagens do Servidor

Ao longo de todo o jogo são também mostradas mensagens por parte do servidor para que seja possível, de uma forma acessível, ver o acesso aos vários predicados de lógica do jogo que estejam a ocorrer ao nível do PROLOG, como podemos ver pelo exemplo a seguir (figura 7).

```

startNewGame                                XMLscene.js:494
[init(humanVhuman)]: success                 MyServer.js:150
[query(board)]: [[[],[],[red],[],[black],[],[black],[],[]],  MyServer.js:150
[[],[],[],[red],[],[green],[],[],[]],[],[],[wild],[],[blue],[],[ivory],[],
[[],[],[ivory],[],[green],[],[red],[],[blue],[],[],[],[wild],[],[blue],[],
[black],[],[],[],[black],[],[ivory],[],[red],[],[ivory],[],[],[black],[],
[green],[],[green],[],[blue],[],[ivory],[],[],[blue],[],[green],[],[ivory],[],
[black],[],[],[],[],[red],[],[blue],[],[red],[],[],[],[],[blue],[],[ivory],[],
[red],[],[ivory],[],[],[],[green],[],[black],[],[green],[],[],[],[],[],
[black],[],[red],[],[],[],[],[],[green],[],[blue],[],[wild],[],[]]]
[query(player)]: player1                     MyServer.js:150
[query(score)]: 20                           MyServer.js:150
[query(player)]: player1                     MyServer.js:150
Picked object: [object Object], with pick id 13 XMLscene.js:182
[query(validMoves,4,2)]: []                  MyServer.js:150
Picked object: [object Object], with pick id 45 XMLscene.js:182
[action(claim,green)]: success               MyServer.js:150

```

Figura 7: Mensagens de Servidor.

Nesta imagem, podemos observar a chamada do predicado que inicia o jogo, seguido do board gerado. Seguidamente, é pedido o score do jogador atual, bem como o seu nome. Depois de selecionar uma peça no jogo, neste caso com id 13, podemos, também observar que não foi retornada nenhuma jogada possível pelo predicado `validMoves`, visto que, a peça selecionada era *wild*, e por isso, não pode ser movida. É depois selecionada a peça 45, que, visto que só existem 44 peças em jogo, sabemos que é uma peça de seleção de cor, o que é comprovado pela chamada do predicado seguinte *claim*, que associa as peças verdes ao *player1*.

Realçamos que a interação com Prolog foi feita seccionando os pedidos ao servidor de acordo com a sua função (*init*, *query* e *action*).