

**Programação em Lógica**

**Relatório Final**

**Barragoon 4**

****

**Leonardo Manuel Gomes Teixeira – up201502848**

**Maria Eduarda Santos Cunha – up201506524**

# Resumo

Resumo sucinto do trabalho com 150 a 250 palavras (problema abordado, objetivo, como foi o problema resolvido/abordado, principais resultados e conclusões).

Índice

[1. Resumo 2](#_Toc498173062)

[2. O Jogo Barragoon 4](#_Toc498173063)

[2.1. Tabuleiro e Peças 4](#_Toc498173064)

[2.2. Objetivo 4](#_Toc498173065)

[2.3. Movimentos 5](#_Toc498173066)

[2.4. Regras 5](#_Toc498173067)

[3. Lógica do Jogo 7](#_Toc498173068)

[2.5. Representação do Estado de Jogo 7](#_Toc498173069)

[2.6. Visualização do Tabuleiro 10](#_Toc498173070)

[2.7. Lista de Jogadas Válidas 11](#_Toc498173071)

[2.8. Execução de Jogadas 11](#_Toc498173072)

[2.9. Avaliação do Tabuleiro 11](#_Toc498173073)

[2.10. Final do Jogo 12](#_Toc498173074)

[2.11. Jogada do Computador 12](#_Toc498173075)

[4. Interface com o Utilizador 13](#_Toc498173076)

[5. Conclusões 14](#_Toc498173077)

[6. Bibliografia 15](#_Toc498173078)

[7. Anexos 16](#_Toc498173079)

# O Jogo Barragoon

O Barragoon é um jogo de estratégia sem qualquer fator de aleatoriedade para 2 jogadores. Foi publicado pela primeira vez a 3 de março de 2014 pela companhia WiWa Spiele UG. As suas regras foram atualizadas pela última vez a 30 de março de 2016.

## Tabuleiro e Peças

O jogo realiza-se num tabuleiro de 9x7 células e os jogadores jogam sempre à vez.

Existem 2 tipos principais de peças: as telhas de cada jogador, brancas ou castanhas, e os barragoons.

* As telhas dos jogadores possuem na sua face um símbolo com 2, 3 ou 4 círculos, relativo ao número de células que podem andar num só movimento (fig.1). Cada jogador começa com 7 telhas: 2 de 2 círculos, 3 de 3 círculos e 2 de 4 círculos;
* O barragoon é a peça central do jogo. É uma peça cúbica, em que cada uma das suas faces possui um símbolo que indica a permissão do jogador de mover a sua peça pela célula em que o barragoon se encontra (fig.2). O jogo começa com 8 barragoons.



**2 células**

**3 células**

**4 células**

Figura 1: Telhas Brancas ou Castanhas



**Sem passagem**

**Uma direção**

**Duas direções**

**Virar à direita**

**Virar à esquerda**

**Todas as direções**

Figura 2: Faces de um Barragoon

## Objetivo

Ambos os jogadores têm de recorrer às suas aptidões táticas para mover as suas telhas e dispor os barragoons de forma a que lhes seja permitido capturar todas as telhas do outro jogador ou, pelo menos, impedir o seu progresso.

O jogo acaba quando um dos jogadores já não consegue mover telhas, porque não possui nenhuma ou por as que tem se encontrarem limitadas por barragoons. O outro é o vencedor.

## Movimentos

Existem 2 tipos de movimentos: full moves e short moves.

* Os full moves correspondem a percorrer x células, de acordo com o número de círculos na telha do jogador (2, 3 ou 4), respetivamente (fig.3);
* Os short moves correspondem a percorrer x-1 células, de acordo com o número de círculos na telha do jogador (2, 3 ou 4, logo movimentos de 1, 2 ou 3 células), respetivamente (fig.4).



Figura 4: Short Move com Telha de 3 Círculos



Figura 3: Full Move com Telha de 4 Círculos

## Regras

* Uma peça é capturada se a peça do oponente terminar na mesma célula que ela (fig.5); s
* Só é possível capturar uma peça durante um full move;
* As telhas com 2 círculos não podem capturar barragoons com o símbolo “todas as direções” virado para cima (fig.6);
* Se um barragoon for capturado, tem de voltar a ser colocado no tabuleiro, numa posição livre à escolha do jogador, com a face levantada para cima que ele preferir;
* Sempre que uma telha é capturada, são adicionados 2 barragoons novos ao tabuleiro, um por cada jogador, e coloca primeiro no tabuleiro o jogador cuja telha foi capturada;
* Nunca se pode mudar um barragoon de posição uma vez colocado;
* Quando se toca numa telha para a mover, não se pode trocar por outra ou voltar atrás;
* Durante um movimento, só se pode efetuar uma mudança de direção uma vez de 90º (fig.7);

Os movimentos só podem ser verticais ou horizontais, nunca na diagonal

Figura 6: Peça de 2 Círculos Não Captura Barragoon com Face “Todas as Direções” Voltada para Cima



Figura 5: Captura de um Barragoon

Figura 7: Movimento Impossível com 2 Mudanças de Direção

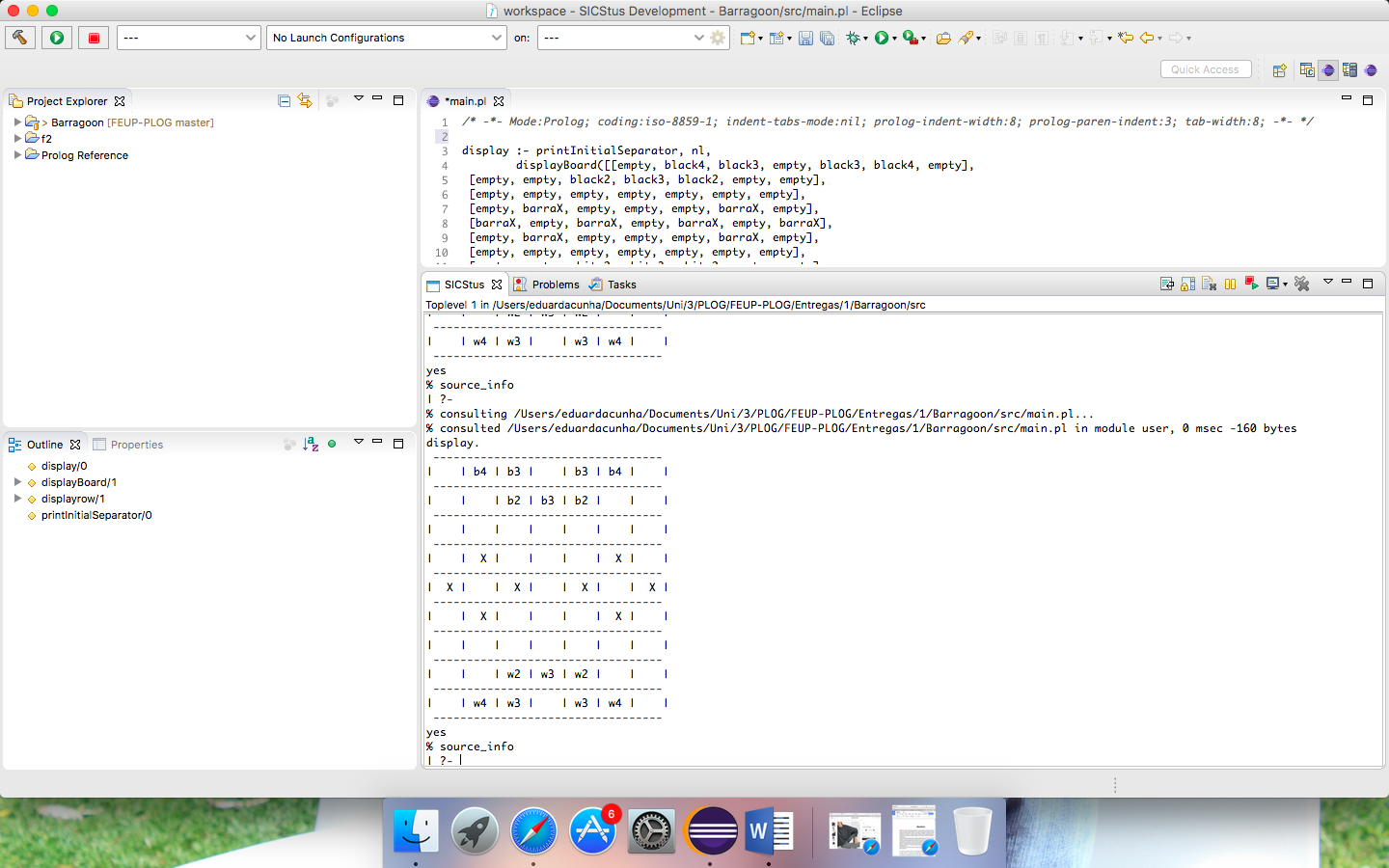


# Lógica do Jogo

Descrever o projeto e implementação da lógica do jogo em Prolog, incluindo a forma de representação do estado do tabuleiro e sua visualização, execução de movimentos, verificação do cumprimento das regras do jogo, determinação do final do jogo e cálculo das jogadas a realizar pelo computador utilizando diversos níveis de jogo.

## Representação do Estado de Jogo

Por questões de simplificação, os barragoons encontram-se aqui representados apenas por, por exemplo, barraX, em vez de bg-‘barraX’. Na prática e em todos os fragmentos de código, referimo-nos a cada barragoon pelo prefixo bg seguido da face voltada para cima.

**Estado Inicial:**

[[empty, b-4, b-3, empty, b-3, b-4, empty],

[empty, empty, b-2, b-3, b-2, empty, empty],

[empty, empty, empty, empty, empty, empty, empty],

[empty, barraX, empty, empty, empty, barraX, empty],

[barraX, empty, barraX, empty, barraX, empty, barraX],

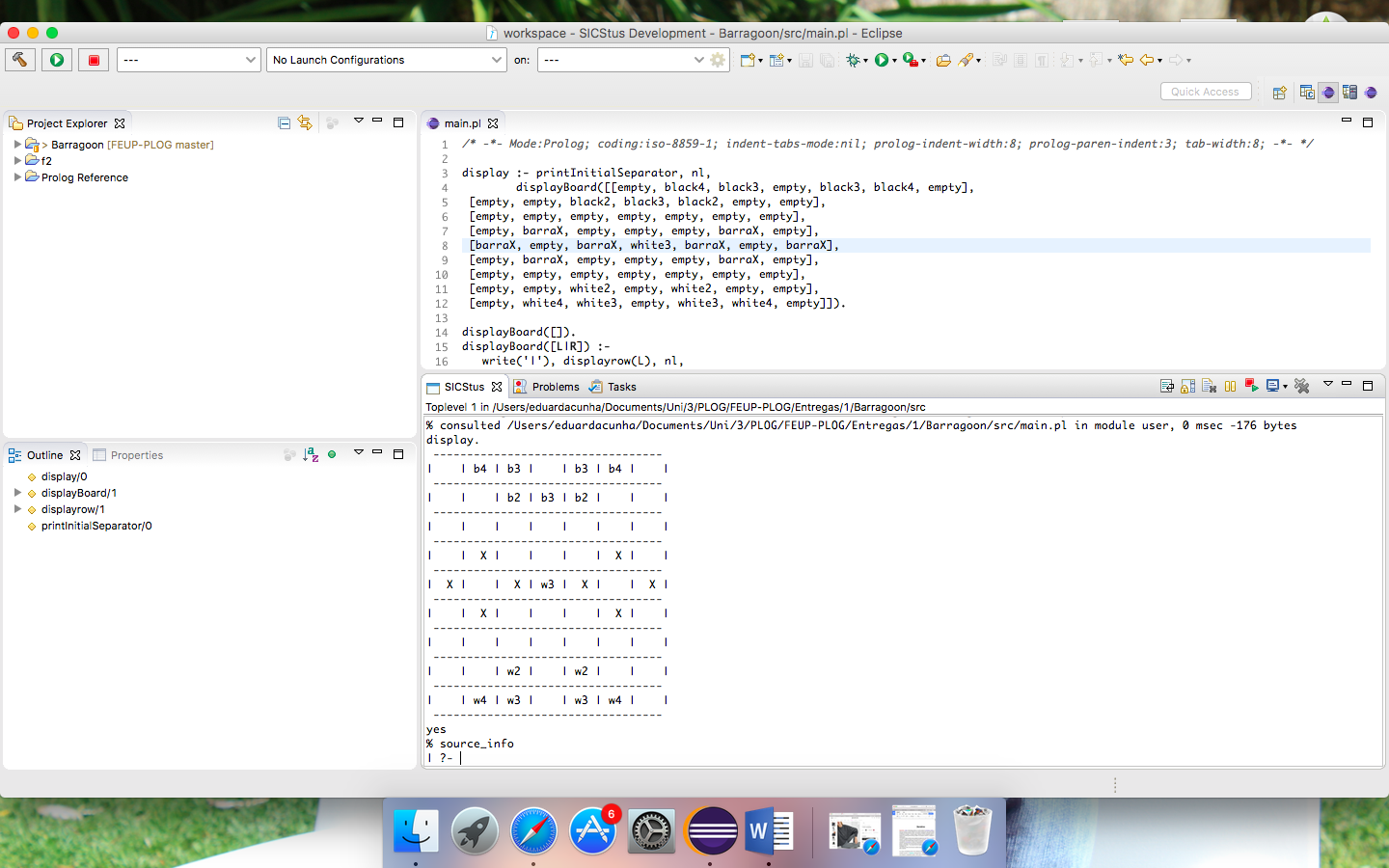
[empty, barraX, empty, empty, empty, barraX, empty],

[empty, empty, empty, empty, empty, empty, empty],

[empty, empty, w-2, w-3, w-2, empty, empty],

 [empty, w-4, w-3, empty, w-3, w-4, empty]]

**Estados intermédios:**

**1.**

[[empty, b-4, b-3, empty, b-3, b-4, empty],

[empty, empty, b-2, b-3, b-2, empty, empty],

[empty, empty, empty, empty, empty, empty, empty],

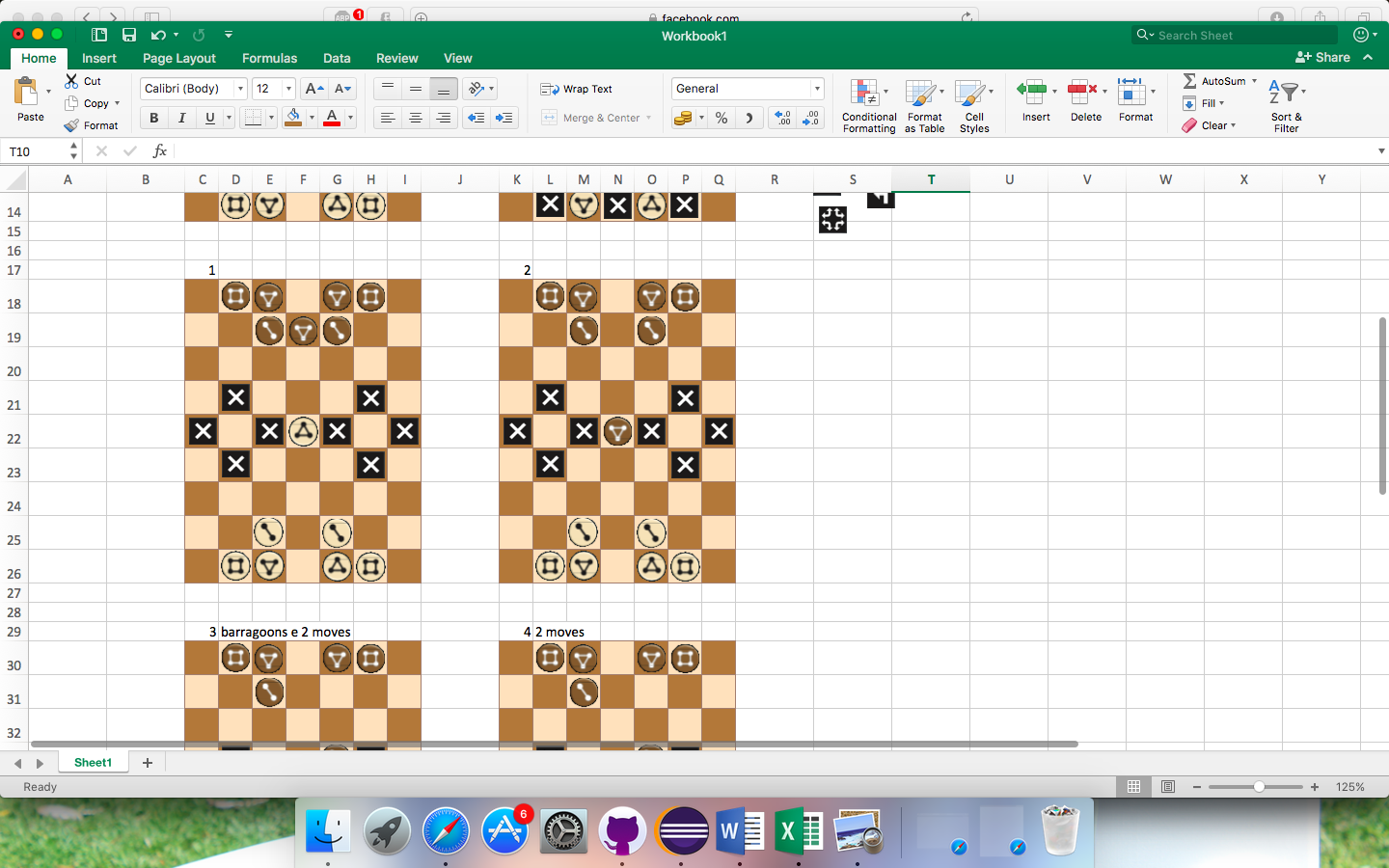
[empty, barraX, empty, empty, empty, barraX, empty],

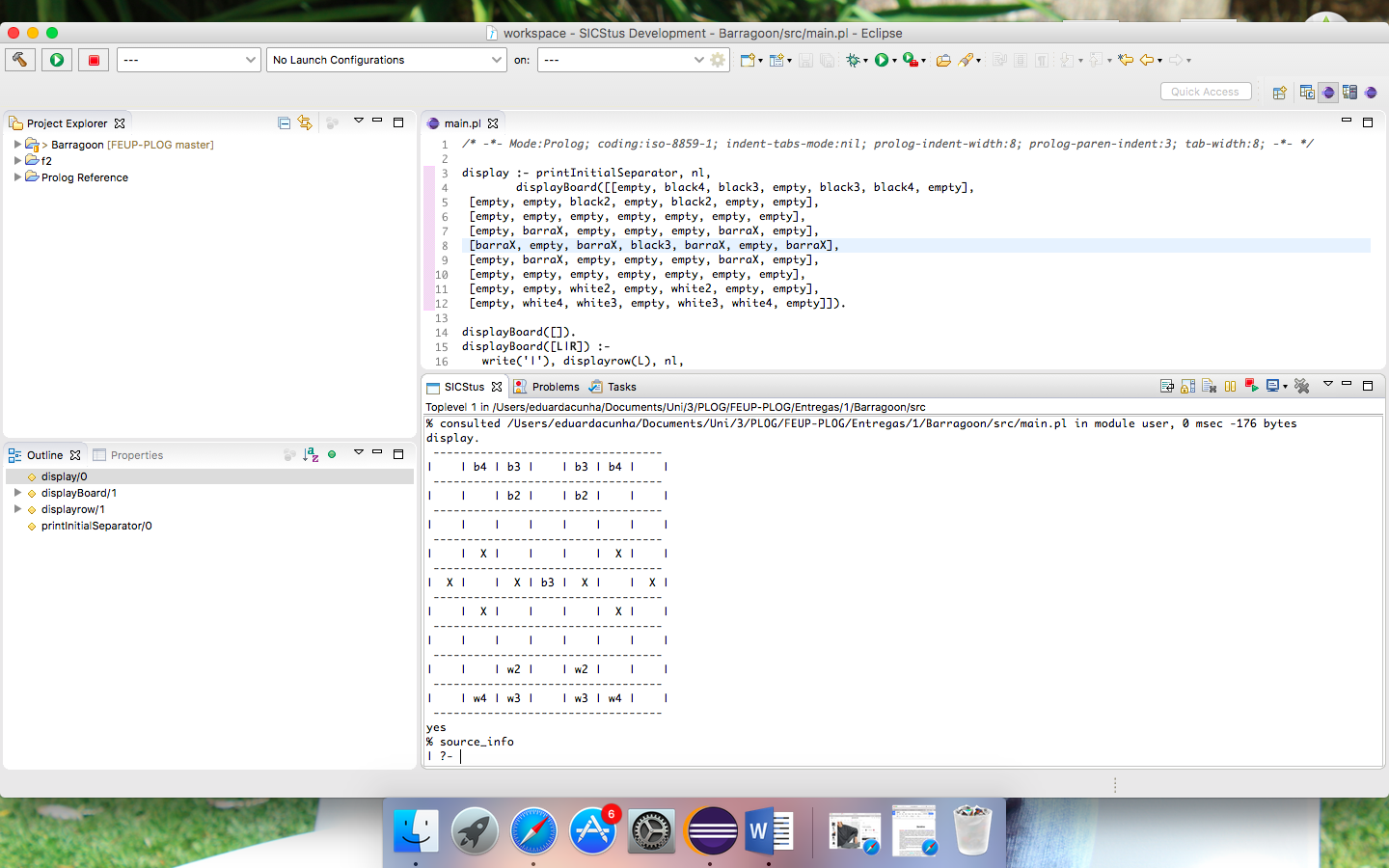
[barraX, empty, barraX, w-3, barraX, empty, barraX],

[empty, barraX, empty, empty, empty, barraX, empty],

[empty, empty, empty, empty, empty, empty, empty],

[empty, empty, w-2, empty, w-2, empty, empty],

 [empty, w-4, w-3, empty, w-3, w-4, empty]]

**2.**

[[empty, b-4, b-3, empty, b-3, b-4, empty],

[empty, empty, b-2, empty, b-2, empty, empty],

[empty, empty, empty, empty, empty, empty, empty],

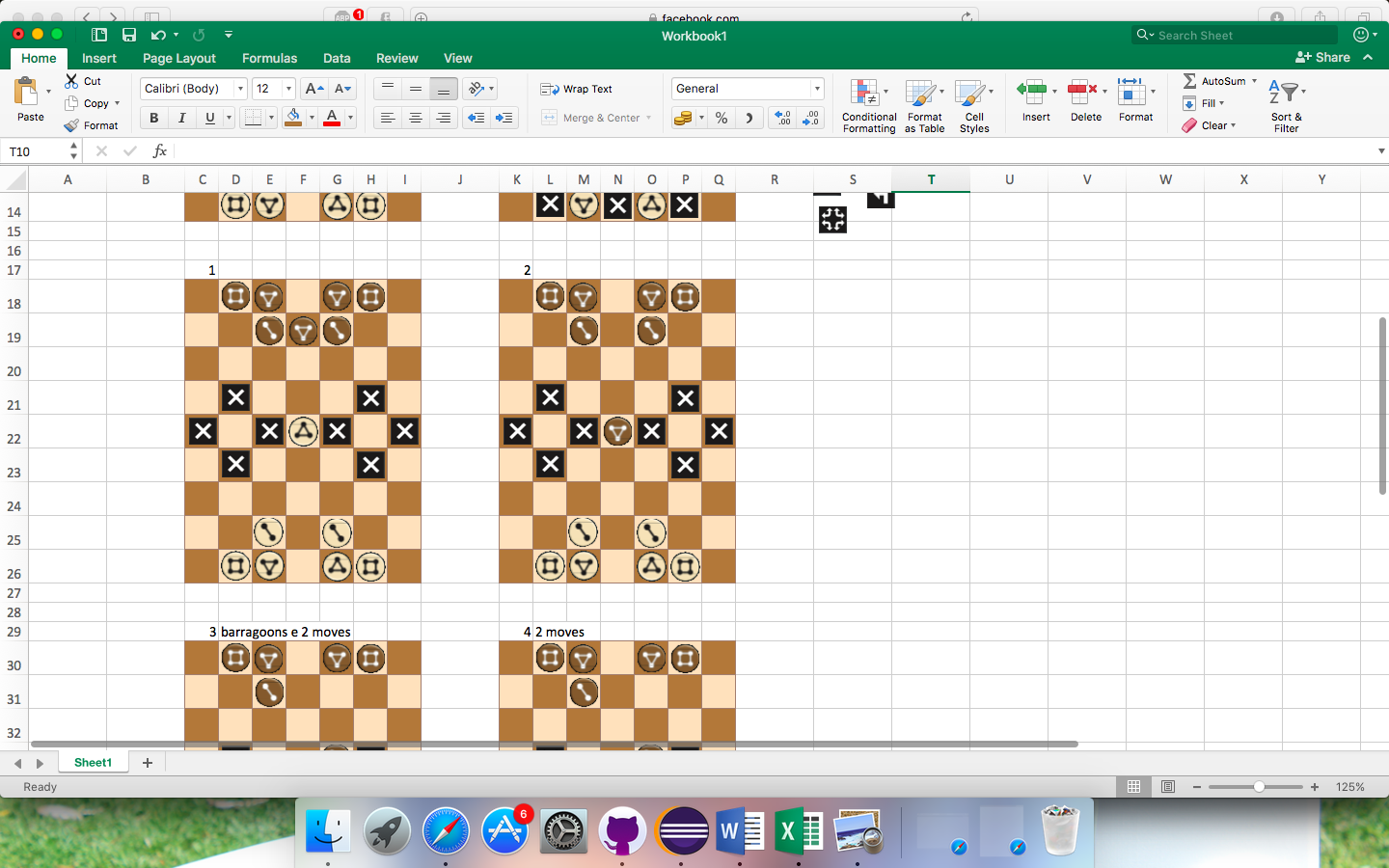
[empty, barraX, empty, empty, empty, barraX, empty],

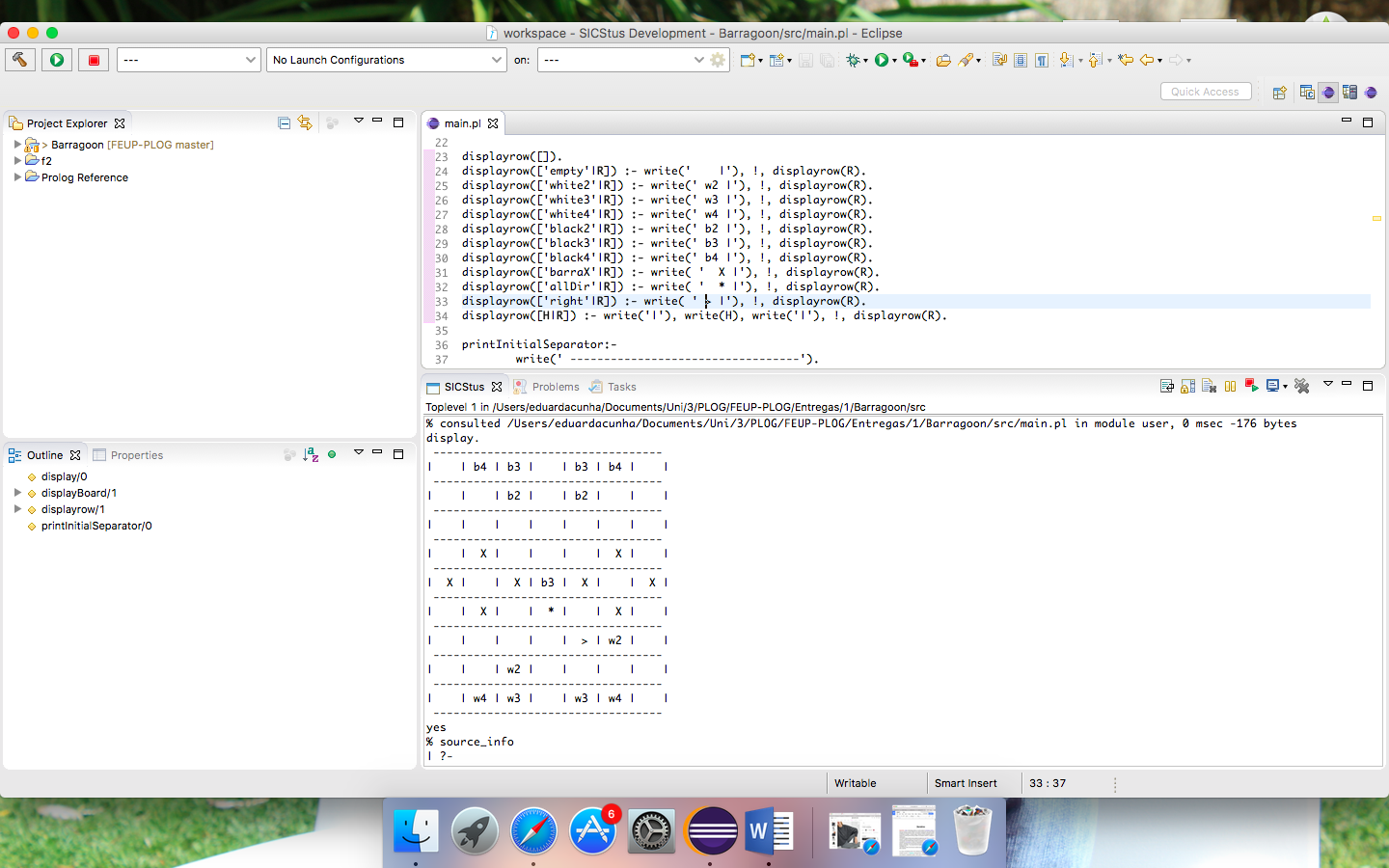
[barraX, empty, barraX, b-3, barraX, empty, barraX],

[empty, barraX, empty, empty, empty, barraX, empty],

[empty, empty, empty, empty, empty, empty, empty],

[empty, empty, w-2, empty, w-2, empty, empty],

**** [empty, w-4, w-3, empty, w-3, w-4, empty]]

**3.**

[[empty, b-4, b-3, empty, b-3, b-4, empty],

[empty, empty, b-2, empty, empty, empty, empty],

[empty, empty, empty, empty, empty, empty, empty],

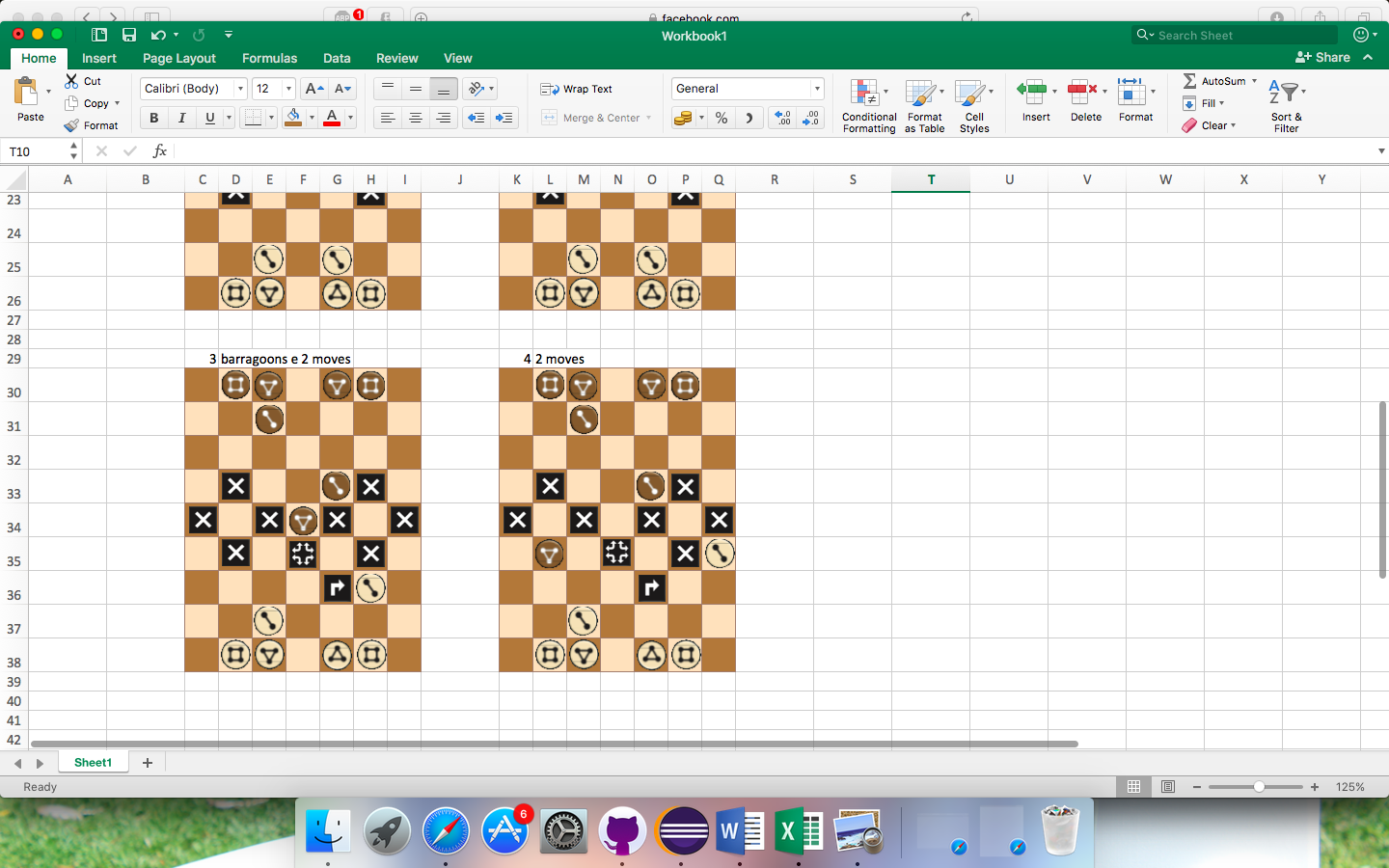
[empty, barraX, empty, empty, b-2, barraX, empty],

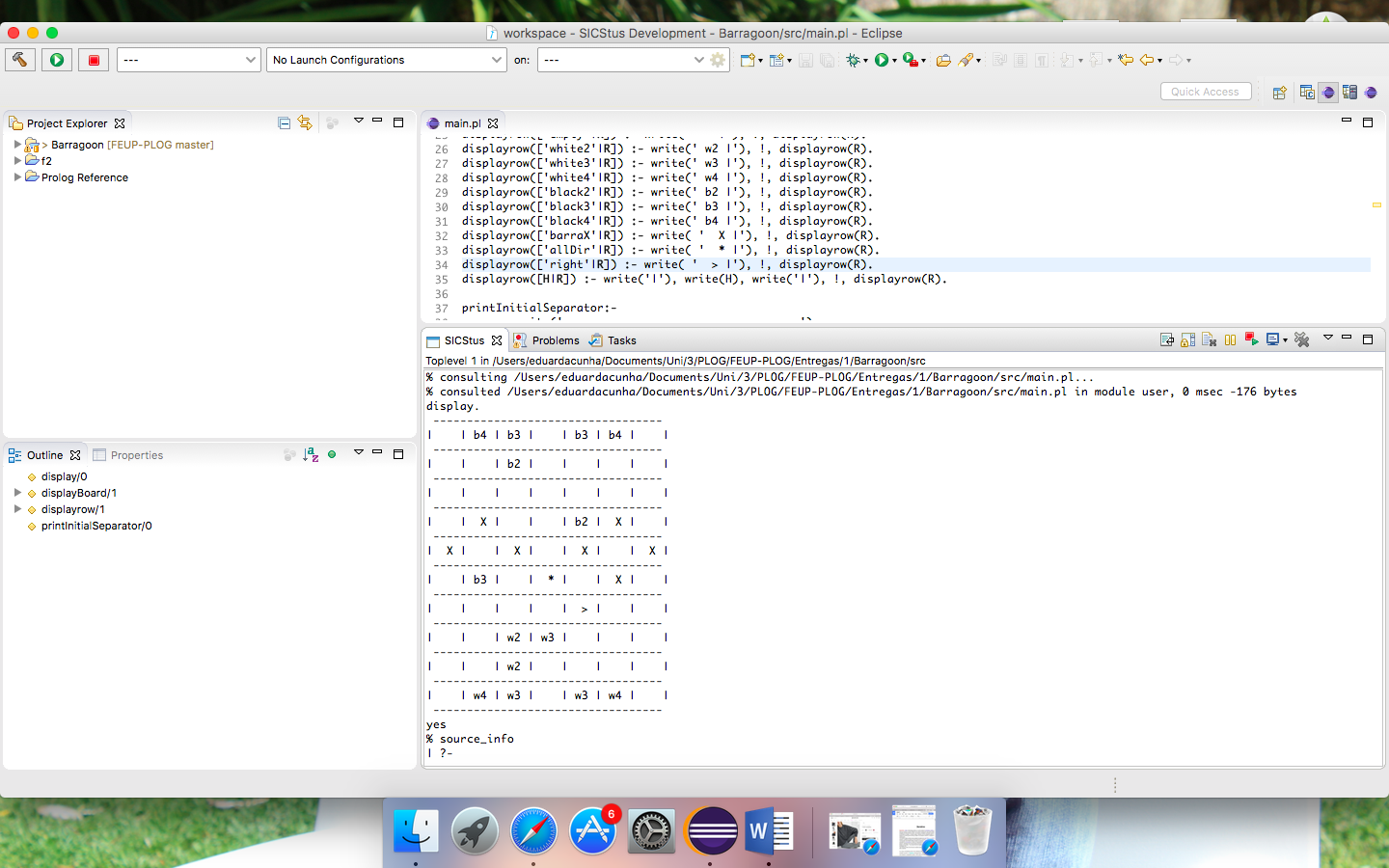
[barraX, empty, barraX, b-3, barraX, empty, barraX],

[empty, barraX, empty, allDir, empty, barraX, empty],

[empty, empty, empty, empty, right, w-2, empty],

[empty, empty, w-2, empty, empty, empty, empty],

 [empty, w-4, w-3, empty, w-3, w-4, empty]]

**4.**

[[empty, b-4, b-3, empty, b-3, b-4, empty],

[empty, empty, b-2, empty, empty, empty, empty],

[empty, empty, empty, empty, empty, empty, empty],

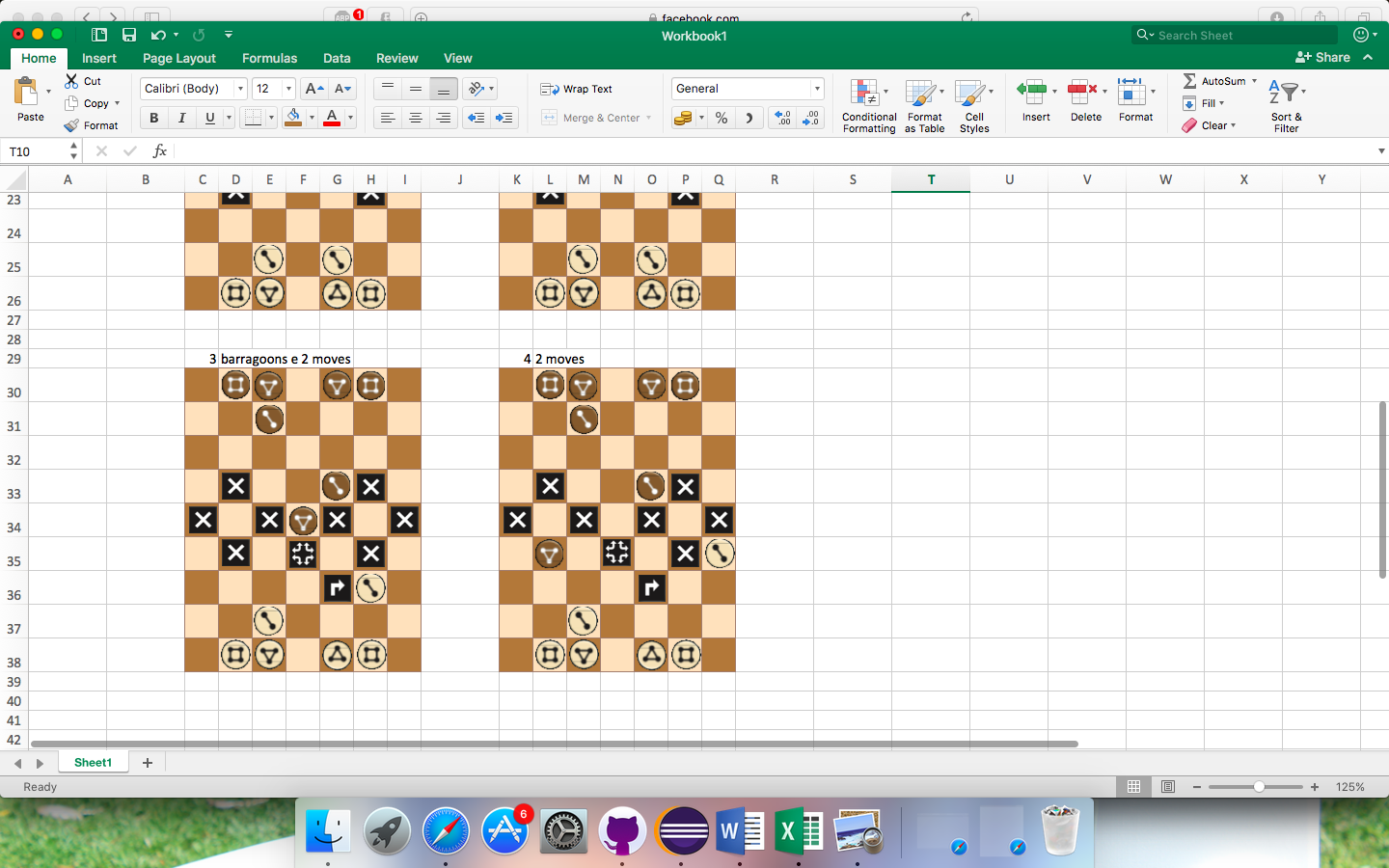
[empty, bg-'barraX', empty, empty, b-2, barraX, empty],

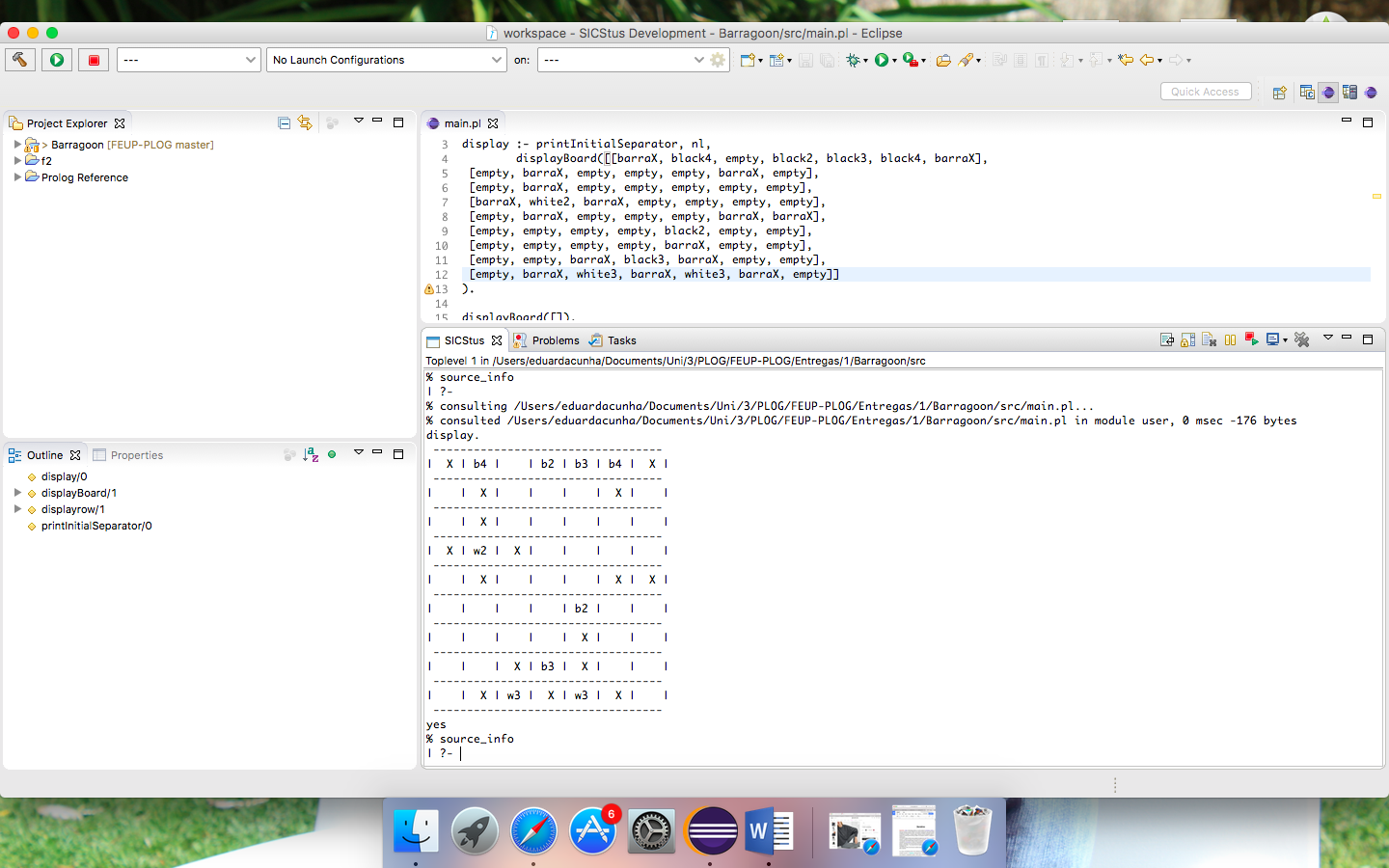
[barraX, empty, barraX, empty, barraX, empty, barraX],

[empty, b-3, empty, allDir, empty, barraX, empty],

[empty, empty, empty, empty, right, empty, empty],

[empty, empty, w-2, w-3, empty, empty, empty],

 [empty, w-4, w-3, empty, w-3, w-4, empty]]

**Estado final:**

[[barraX, b-4, empty, b-2, b-3, b-4, barraX],

[empty, barraX, empty, empty, empty, barraX, empty],

[empty, barraX, empty, empty, empty, empty, empty],

[barraX, w-2, barraX, empty, empty, empty, empty],

[empty, barraX, empty, empty, empty, barraX, barraX],

[empty, empty, empty, empty, b-2, empty, empty],

[empty, empty, empty, empty, barraX, empty, empty],

[empty, empty, barraX, b-3, barraX, empty, empty],

 [empty, barraX, w-3, barraX, w-3, barraX, empty]]

## Visualização do Tabuleiro

1. initialBoard(
2. [[empty, b-*4*, b-*3*, empty, b-*3*, b-*4*, empty],
3. [empty, empty, b-*2*, b-*3*, b-*2*, empty, empty],
4. [empty, empty, empty, empty, empty, empty, empty],
5. [empty, bg-'barraX', empty, empty, empty, bg-'barraX', empty],
6. [bg-'barraX', empty, bg-'barraX', empty, bg-'barraX', empty, bg-'barraX'],
7. [empty, bg-'barraX', empty, empty, empty, bg-'barraX', empty],
8. [empty, empty, empty, empty, empty, empty, empty],
9. [empty, empty, w-*2*, w-*3*, w-*2*, empty, empty],
10. [empty, w-*4*, w-*3*, empty, w-*3*, w-*4*, empty]])**.**
11. displayGame(**Game**) **:-**
12. getBoard(Game, **Board**),
13. getCurrentPlayer(Game, **Player**),
14. clearScreen,
15. displayPlayerTurn(Player),
16. lettersAxis,nl,
17. horizontalBorder, nl,
18. numbersAxis(**RowNumbers**),
19. displayBoard(Board, RowNumbers), nl**.**
20. displayBoard([], [])**.**
21. displayBoard([**RowToDisplay**|**RemainingBoard**],
22. [**RowToDisplayNumber**|**RemainingRowNumbers**]) **:-**
23. translate([RowToDisplayNumber]),
24. translate(RowToDisplay),border, nl,
25. horizontalBorder, nl,
26. displayBoard(RemainingBoard, RemainingRowNumbers)**.**
27. *% -- Board Translation --*
28. translate([])**.**
29. translate(['empty'|**R**]) **:-** border, write(' '), !, translate(R)**.**
30. translate([w-*2*|**R**]) **:-** border, write(' w2 '), !, translate(R)**.**
31. translate([w-*3*|**R**]) **:-** border, write(' w3 '), !, translate(R)**.**
32. translate([w-*4*|**R**]) **:-** border, write(' w4 '), !, translate(R)**.**
33. translate([b-*2*|**R**]) **:-** border, write(' b2 '), !, translate(R)**.**
34. translate([b-*3*|**R**]) **:-** border, write(' b3 '), !, translate(R)**.**
35. translate([b-*4*|**R**]) **:-** border, write(' b4 '), !, translate(R)**.**
36. translate([bg-'barraX'|**R**]) **:-** border, write(' X '), !, translate(R)**.**
37. translate([bg-'allDir'|**R**]) **:-** border, write(' + '), !, translate(R)**.**
38. translate([bg-'1DirU'|**R**]) **:-** border, write(' V '), !, translate(R)**.**
39. translate([bg-'1DirD'|**R**]) **:-** border, write(' A '), !, translate(R)**.**
40. translate([bg-'1DirL'|**R**]) **:-** border, write(' <= '), !, translate(R)**.**
41. translate([bg-'1DirR'|**R**]) **:-** border, write(' => '), !, translate(R)**.**
42. translate([bg-'2DirH'|**R**]) **:-** border, write(' X '), !, translate(R)**.**
43. translate([bg-'2DirV'|**R**]) **:-** border, write(' X '), !, translate(R)**.**
44. translate([bg-'right'|**R**]) **:-** border, write(' > '), !, translate(R)**.**
45. translate([bg-'left'|**R**]) **:-** border, write(' < '), !, translate(R)**.**
46. translate(['um'|**R**]) **:-** write('1'), !, translate(R)**.**
47. translate(['dois'|**R**]) **:-** write('2'), !, translate(R)**.**
48. translate(['tres'|**R**]) **:-** write('3'), !, translate(R)**.**
49. translate(['quatro'|**R**]) **:-** write('4'), !, translate(R)**.**
50. translate(['cinco'|**R**]) **:-** write('5'), !, translate(R)**.**
51. translate(['seis'|**R**]) **:-** write('6'), !, translate(R)**.**
52. translate(['sete'|**R**]) **:-** write('7'), !, translate(R)**.**
53. translate(['oito'|**R**]) **:-** write('8'), !, translate(R)**.**
54. translate(['nove'|**R**]) **:-** write('9'), !, translate(R)**.**
55. *% -- Board Axis --*
56. numbersAxis([um, dois, tres, quatro, cinco, seis, sete, oito, nove])**.**
57. lettersAxis **:-** write(' A B C D E F G')**.**
58. *% -- Board Borders --*
59. horizontalBorder **:-** write(' ----------------------------------')**.**
60. border **:-** write('|')**.**

## Lista de Jogadas Válidas

Obtenção de uma lista de jogadas possíveis.

## Execução de Jogadas

Validação e execução de uma jogada num tabuleiro, obtendo o novo estado do jogo.

## Avaliação do Tabuleiro

Avaliação do estado do jogo, que permitirá comparar a aplicação das diversas jogadas disponíveis.

## Final do Jogo

Verificação do fim do jogo, com identificação do vencedor.

## Jogada do Computador

Escolha da jogada a efetuar pelo computador, dependendo do nível de dificuldade.

# Interface com o Utilizador

Descrever o módulo de interface com o utilizador em modo de texto.

# Conclusões

Que conclui este projeto? Como poderia melhorar o trabalho desenvolvido?

# Bibliografia

1. https://www.youtube.com/watch?v=qG1i0\_sn\_FI
2. https://boardgamegeek.com/boardgame/157779/barragooN
3. http://www.barragoon.de/bsp/BARRAGOON\_en.pdf
4. https://stackoverflow.com
5. http://www.swi-prolog.org

# Anexos

main.pl

Interface.pl

Logic.pl

Utilities.pl