

GARDENS OF IO

Relatório Final



Mestrado Integrado em Engenharia Informática
e Computação

Grupo Gardens_of_Io_1:

Filipe Ribeiro – 201104129

Ricardo Neves – 201405868

Faculdade de Engenharia da Universidade do Porto Rua Roberto Frias, sn,
4200-465 Porto, Portugal

14 de Novembro de 2016

Resumo

O jogo não está implementado por completo. Faltam verificações e a finalização do modo individual.

No entanto, é possível, caso sejam seguidas todas as instruções, jogar o jogo por pleno, isto é, se todas as coordenadas forem inseridas corretamente, o jogo corre e chega ao fim sem problemas.

A equipa defrontou-se com dificuldades inesperadas mas tentou sobrepor-se sempre que tal. Neste jogo podemos jogar com o máximo de 5 jogadores e a cada jogada ou utilizámos uma ação especial ou plantámos uma flor.

Índice

1.Introdução.....	4
2.O Jogo “Gardens of Io”	5,6
3.Lógica do Jogo.....	7
3.1 Representação do Estado do Jogo.....	7,8
3.2 Visualização do Tabuleiro.....	9,10
3.3 Lista de Jogadas Válidas	10
3.4 Execução de Jogadas.....	10
3.5 Avaliação do Tabuleiro.....	11
3.6 Final do Jogo	11
3.7Jogada do Computador	11
4.Interface com o Utilizador	12,13,14
5.Conclusões.....	14
6.Bibliografia	15
7.Anexo.....	15

1.Introdução

Em prol da unidade curricular de Programação em Lógica do curso Mestrado Integrado em Engenharia Informática e Computação, fizemos este projeto que consiste na elaboração de um jogo de tabuleiro em PROLOG. Com uma vasta lista de opções, o grupo decidiu fazer o jogo “Gardens of Io” pela curiosidade que despertou na sua descrição e pela complexidade apresentada.

O jogo é pouco ortodoxo quando comparado com o famoso e tradicional jogo das damas, no entanto, por ser mais complexo, também requer uma maior concentração e habilidade estratégica para conseguir sobressair sobre os adversários. Uma das particularidades deste jogo é o anexo, no tabuleiro de uma linha de ações especiais que contém 5 ações possíveis de utilizar apenas 1 vez por jogo que pretendem abolir a “perda prematura” do jogo, isto é, o jogo pode ser ganho por qualquer um dos jogadores até que este acabe.

Quanto à restante estrutura do relatório:

- História e regras do jogo elaborado;
- Lógica e descrição do projeto;
- Explicação da interface do jogo e a relação entre o programa e o utilizador, jogabilidade e visualização;
- Conclusão explicando as maiores dificuldades;
- Bibliografia
- Anexos

2. O Jogo “Gardens of Io”

Este jogo tem como cena, uma das luas de Júpiter, Io. Um grupo de marcianos viajou até lá com o objetivo de colonizar e embelezar a lua, através de um aperfeiçoamento das técnicas de jardinagem. Estas técnicas consistem em plantar flores, com sementes de crescimento instantâneo, de tal ordem que evitam passar por cima delas, evitando que elas se estraguem.

“Gardens of Io” é jogado no seguinte tabuleiro:

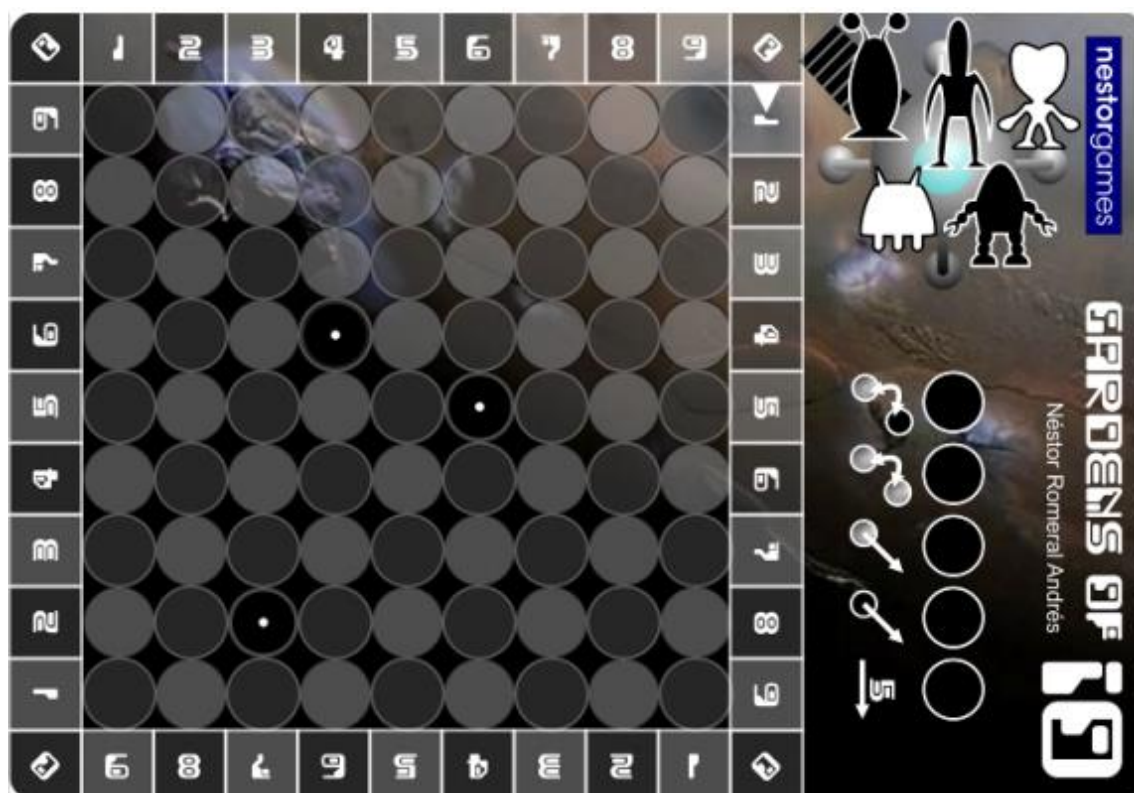


Figura 1 - Tabuleiro de jogo

O jogo pode ser jogado por 2-5 jogadores. Cada jogador escolhe um marciano diferente e os jogadores colocam 3 árvores no tabuleiro pela ordem de jogo (na imagem anterior podem ser observadas as representações de 3 árvores já em jogo). Dependendo do número de jogadores, os jogadores deverão tirar à sorte um determinado número de flores. Há 6 cores diferentes de flores: Branco, Amarelo, Azul, Roxo, Verde e Vermelho.

No seu turno cada jogador deverá, pela seguinte ordem:

1- O jogador poderá, uma vez por jogo, utilizar uma ação especial que ainda não tenha sido utilizada (as ações especiais são os 5 círculos ao lado do tabuleiro na imagem anterior). As ações especiais são todas diferentes. No caso de só haver dois jogadores, cada jogador poderá usar duas ações diferentes.

2- O jogador poderá colocar no tabuleiro qualquer flor que tenha na mão, desde que o caminho para a posição pretendida não esteja bloqueado por uma árvore (caminhos em linha reta ou na diagonal apenas).

3- O jogador ganha o número de pontos correspondente à quantidade de flores, da mesma cor que jogou, que estão juntas ortogonalmente.

4- O jogador move o seu marciano o número de casas correspondente ao número de pontos efetuados nessa jogada. Só contam as casas livres, ou seja, caso o jogador marque 1 ponto, mas na casa seguinte esteja um marciano diferente, o jogador avança duas casas. Caso o jogador complete uma volta, este pode pegar numa flor que já esteja no tabuleiro, exceto numa cor que já tenha sido esgotada, e pode adicioná-la à sua mão.

5- Se o jogador tiver jogado a última flor de uma certa cor, ou seja, já estão todas no tabuleiro, o jogador tem direito a um turno extra.

O jogo termina quando um jogador joga todas as suas flores, ou quando o jogador que está mais adiantado consegue dar uma volta de avanço ao mais atrasado. Ganha o jogador que estiver mais avançado, sendo que o número de voltas dadas conta.

3. Lógica do Jogo

3.1. Representação do Estado do Jogo

Representação do estado inicial do tabuleiro:

```
[ ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0'],
  ['9', empty, empty, empty, empty, empty, empty, empty, empty, empty, '1'],
  ['8', empty, empty, empty, empty, empty, empty, empty, empty, empty, '2'],
  ['7', empty, empty, empty, empty, empty, empty, empty, empty, empty, '3'],
  ['6', empty, empty, empty, empty, empty, empty, empty, empty, empty, '4'],
  ['5', empty, empty, empty, empty, empty, empty, empty, empty, empty, '5'],
  ['4', empty, empty, empty, empty, empty, empty, empty, empty, empty, '6'],
  ['3', empty, empty, empty, empty, empty, empty, empty, empty, empty, '7'],
  ['2', empty, empty, empty, empty, empty, empty, empty, empty, empty, '8'],
  ['1', empty, empty, empty, empty, empty, empty, empty, empty, empty, '9'],
  ['0', '9', '8', '7', '6', '5', '4', '3', '2', '1', '0'] ]
```

[action1, action2, action3, action4, action5]

0	1	2	3	4	5	6	7	8	9	0
9										1
8										2
7										3
6										4
5										5
4										6
3										7
2										8
1										9
0	9	8	7	6	5	4	3	2	1	0
A1	A2	A3	A4	A5						

Figura 2 - Tabuleiro inicial

Representação de um possível estado intermédio do tabuleiro:

```
[ ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0'],
  ['9', empty, empty, empty, empty, empty, empty, empty, empty, empty, '1'],
  ['8', empty, tree, empty, empty, empty, green, empty, empty, empty, '2'],
  ['7', empty, empty, empty, empty, empty, green, empty, blue, empty, '3'],
  [pl1, empty, empty, empty, tree, empty, empty, empty, empty, empty, '4'],
  ['5', empty, empty, empty, empty, empty, green, empty, empty, empty, '5'],
  ['4', empty, empty, red, empty, empty, empty, empty, empty, empty, '6'],
  ['3', empty, empty, red, empty, empty, tree, empty, empty, empty, '7'],
  ['2', empty, empty, empty, empty, empty, empty, empty, empty, empty, '8'],
  ['1', empty, empty, empty, empty, empty, empty, empty, empty, empty, '9'],
  ['0', pl2, pl3, '7', '6', '5', '4', '3', '2', '1', '0'] ]
```

[action1, **empty**, **empty**, action4, action5]

0	1	2	A	4	5	6	7	8	9	0
9								#		1
8						*				2
7			X							3
6	\$	\$						*		4
5						X				5
4										6
3				X						7
2						&				8
1						&				9
0	9	8	7	6	5	4	3	2	1	0

A1	A2	X	A4	X
----	----	---	----	---

Figura 3 - Tabuleiro intermédio, algumas jogadas efetuadas, símbolos correspondem a flores e árvores.

3.2. Visualização do Tabuleiro

Aqui encontra-se a porção de código que vai fazer com que o tabuleiro seja visualizado na consola:

%Trocar variáveis identificadoras pelos símbolos
correspondentes%

```
getCellSymbol(emptyCell, ' ').  
getCellSymbol(number0Cell, '0').  
getCellSymbol(number1Cell, '1').  
getCellSymbol(number2Cell, '2').  
getCellSymbol(number3Cell, '3').  
getCellSymbol(number4Cell, '4').  
getCellSymbol(number5Cell, '5').  
getCellSymbol(number6Cell, '6').  
getCellSymbol(number7Cell, '7').  
getCellSymbol(number8Cell, '8').  
getCellSymbol(number9Cell, '9').  
getCellSymbol(action1Cell, 'V').  
getCellSymbol(action2Cell, 'W').  
getCellSymbol(action3Cell, 'X').  
getCellSymbol(action4Cell, 'Y').  
getCellSymbol(action5Cell, 'Z').
```

%Visualizar o tabuleiro%

printBoard([]):-

```
    nl, nl,  
    printRow(Actions).
```

printBoard([Head | Tail]):-

```
    write(' | '),  
    printRow(Head), nl,  
    printBoard(Tail).
```

printRow([]).

printRow([Head | Tail]):-

```
    getCellSymbol(Head, Piece),  
    write(Piece), write(' | '),  
    printRow(Tail).
```

3.3. Lista de Jogadas Válidas

Criada a função “validMoves” mas, não está corretamente implementada.

3.4. Execução de Jogadas

Através do menu de jogadas “turnMenu”, é nos dado as opções de jogada para usar uma ação especial ou para plantar uma flor. Caso seja para plantar uma flor, é chamada a função “flowerPlacementMenu(Players, Turn, Board, NBoard, PlayersInfo, NPlayersInfo)” que poe uma planta no local pretendido.

Caso seja escolhida a opção de uma ação, é chamada a função “actionMenu(Players, Turn, Board, NBoard, PlayersInfo, NPlayersInfo)” que permite visualizar as ações possíveis e depois executá-las.

3.5. Avaliação do Tabuleiro

O grupo achou muito difícil de implementar esta funcionalidade visto que é impossível de determinar quem está “à frente” no jogo.

Eventualmente seria possível apenas comparar os “Scores” do jogador e do computador mas estes podem mudar subitamente.

A função `checkHighestScore(Players,PlayersInfo)` poderá correr a lista de jogadores e por cada jogador ver quantas voltas foram dadas, para assim contar o número de pontos desse jogador e comparar com a do jogador seguinte. No entanto, caso fosse implementada um tipo de “inteligência” para o computador, seria a de tentar “adivinhar” a próxima jogada do adversário pela contagem de flores consecutivas e colocar uma na casa diagonal de 1 casa de distância, isto porque, a jogada do adversário seria na continuação da sequência de flores então, na jogada seguinte do computador, em vez de conseguir apenas flores + 1 pontos, conseguiria flores + 3 pontos.

3.6. Final do Jogo

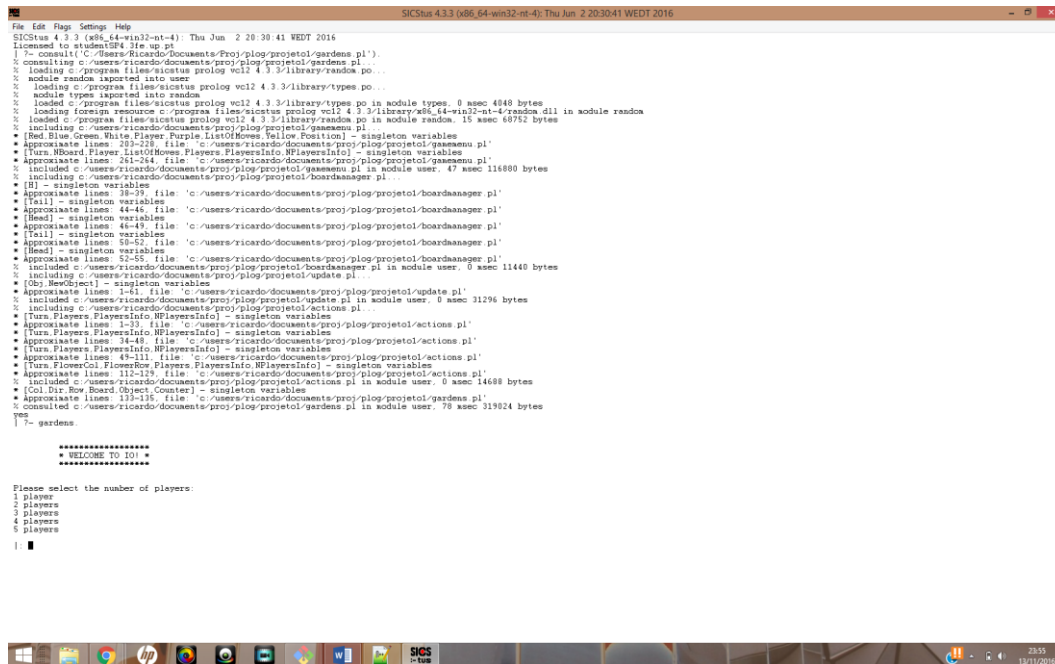
O final do jogo é determinado pelo número de flores do jogador durante uma volta, caso o jogador fique sem flores enquanto está a meio de uma “lap”, o jogo acaba e ganha o jogador com maior número de voltas.

3.7. Jogada do Computador

A jogada do computador é feita apenas por valores aleatórios. Após gerar uma possível jogada, verifica se é válida e, caso seja, joga, caso não seja, gera novas jogadas até encontrar uma válida.

4. Interface com o Utilizador

Entrado no menu principal, ao correr a função “gardens”:

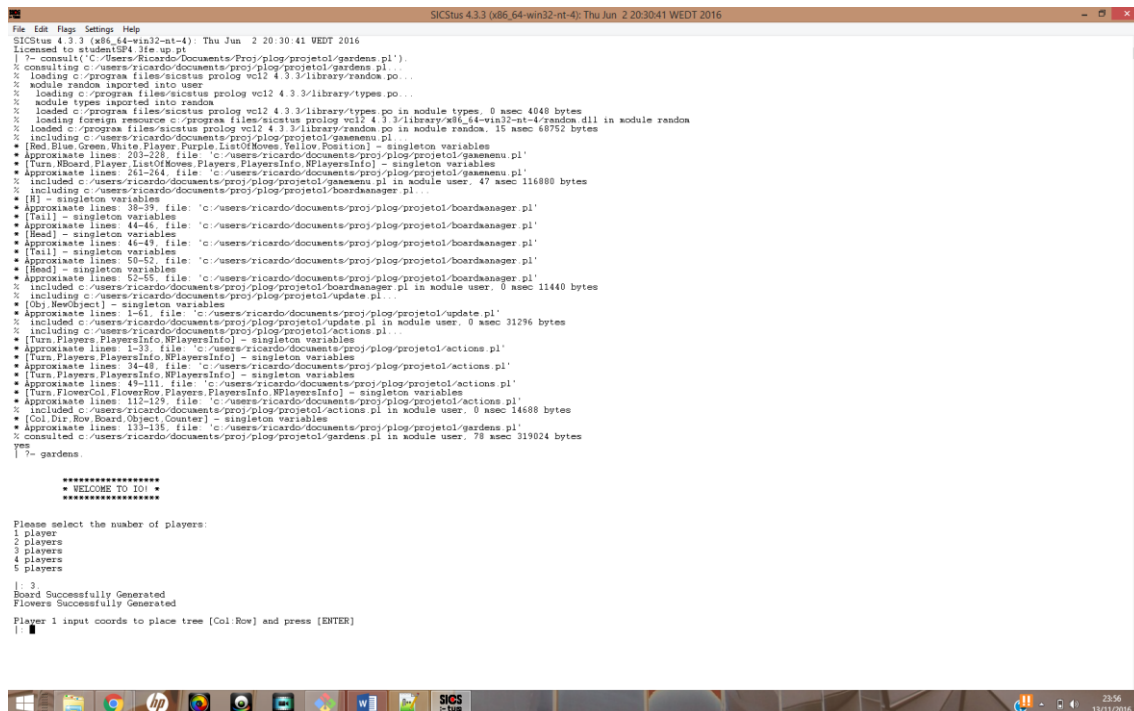


```
File Edit Flags Settings Help
SICStus 4.3.3 (x86_64-win32-nt-4): Thu Jun 2 20:30:41 WEDT 2016
Licensed to student194 Jfe up pt
?- consult('C:/Users/Ricardo/Documents/Proj/plog/projetol/gardens.pl').
% consulting c:/Users/Ricardo/Documents/Proj/plog/projetol/gardens.pl...
% loading c:/program files/sicstus prolog vcl2 4.3.3/library/random.pl...
% module random imported into user
% loading c:/program files/sicstus prolog vcl2 4.3.3/library/types.pl...
% module types imported into random
% loading foreign resource c:/program files/sicstus prolog vcl2 4.3.3/library/x86_64-win32-nt-4/random.dll in module random
% loading c:/program files/sicstus prolog vcl2 4.3.3/library/random.pl in module random. 15 msec 68752 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl...
% [Red.Blue.Green.White.Player.Purple.ListOfMoves.Yellow.Position] - singleton variables
% Approximate lines: 203-226, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl'
% [Turn.Board.Player.ListOfMoves.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 261-264, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl in module user. 47 msec 116880 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl...
% [N] - singleton variables
% Approximate lines: 38-39, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Tail] - singleton variables
% Approximate lines: 44-46, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Head] - singleton variables
% Approximate lines: 44-46, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Tail] - singleton variables
% Approximate lines: 50-52, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Head] - singleton variables
% Approximate lines: 52-55, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl in module user. 0 msec 11440 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/update.pl...
% [Obj.NewObject] - singleton variables
% Approximate lines: 1-41, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/update.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/update.pl in module user. 0 msec 31296 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl...
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 1-13, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 34-34, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 49-111, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 112-129, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl in module user. 0 msec 14688 bytes
% [Col.Dir.Row.Board.Object.Counter] - singleton variables
% Approximate lines: 133-136, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% consulting c:/Users/Ricardo/Documents/Proj/plog/projetol/gardens.pl in module user. 78 msec 319024 bytes
yes
?- gardens.

*****
* WELCOME TO IO! *
*****

Please select the number of players:
1 player
2 players
3 players
4 players
5 players
|
```

Após a escolha do número de jogadores:



```
File Edit Flags Settings Help
SICStus 4.3.3 (x86_64-win32-nt-4): Thu Jun 2 20:30:41 WEDT 2016
Licensed to student194 Jfe up pt
?- consult('C:/Users/Ricardo/Documents/Proj/plog/projetol/gardens.pl').
% consulting c:/Users/Ricardo/Documents/Proj/plog/projetol/gardens.pl...
% loading c:/program files/sicstus prolog vcl2 4.3.3/library/random.pl...
% module random imported into user
% loading c:/program files/sicstus prolog vcl2 4.3.3/library/types.pl...
% module types imported into random
% loading foreign resource c:/program files/sicstus prolog vcl2 4.3.3/library/x86_64-win32-nt-4/random.dll in module random
% loading c:/program files/sicstus prolog vcl2 4.3.3/library/random.pl in module random. 15 msec 68752 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl...
% [Red.Blue.Green.White.Player.Purple.ListOfMoves.Yellow.Position] - singleton variables
% Approximate lines: 203-226, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl'
% [Turn.Board.Player.ListOfMoves.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 261-264, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/gasmenu.pl in module user. 47 msec 116880 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl...
% [N] - singleton variables
% Approximate lines: 38-39, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Tail] - singleton variables
% Approximate lines: 44-46, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Head] - singleton variables
% Approximate lines: 44-46, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Tail] - singleton variables
% Approximate lines: 50-52, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% [Head] - singleton variables
% Approximate lines: 52-55, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/boardmanager.pl in module user. 0 msec 11440 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/update.pl...
% [Obj.NewObject] - singleton variables
% Approximate lines: 1-41, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/update.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/update.pl in module user. 0 msec 31296 bytes
% including c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl...
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 1-13, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 34-48, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 49-111, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% [Turn.Players.PlayersInfo.NPlayersInfo] - singleton variables
% Approximate lines: 112-129, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% included c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl in module user. 0 msec 14688 bytes
% [Col.Dir.Row.Board.Object.Counter] - singleton variables
% Approximate lines: 133-136, file: 'c:/Users/Ricardo/Documents/Proj/plog/projetol/actions.pl'
% consulting c:/Users/Ricardo/Documents/Proj/plog/projetol/gardens.pl in module user. 78 msec 319024 bytes
yes
?- gardens.

*****
* WELCOME TO IO! *
*****

Please select the number of players:
1 player
2 players
3 players
4 players
5 players
| 3
Board Successfully Generated
Flowers Successfully Generated
Player 1 input coords to place tree [Col.Row] and press [ENTER]
|
```

Após a colocação de todas as árvores no tabuleiro:

```

SICStus 4.3.3 (x86_64-win32-nt-4): Thu Jun 2 20:30:41 WEDT 2016
File Edit Flags Settings Help

*****
* WELCOME TO IOI *
*****

Please select the number of players:
1 player
2 players
3 players
4 players
5 players

|: 3
Board Successfully Generated
Flowers Successfully Generated

Player 1 input coords to place tree [Col:Row] and press [ENTER]
|: 1:1

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7
6
5
4
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 2 input coords to place tree [Col:Row] and press [ENTER]
|: 5:5

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7
6
5 T
4
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 3 input coords to place tree [Col:Row] and press [ENTER]
|: 7:7

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7
6
5
4 T
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 1 turn. You have completed 0 laps. Please select an option.
1: PLACE_FLOWER
2: USE_AN_ACTION
3: VALID_MOVES

|:

```

Após a escolha de plantar uma flor e uma ação especial:

```

SICStus 4.3.3 (x86_64-win32-nt-4): Thu Jun 2 20:30:41 WEDT 2016
File Edit Flags Settings Help

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7
6
5
4
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 2 input coords to place tree [Col:Row] and press [ENTER]
|: 3:3

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7 T
6
5
4
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 3 input coords to place tree [Col:Row] and press [ENTER]
|: 7:7

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7 T
6
5
4 T
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 1 turn. You have completed 0 laps. Please select an option.
1: PLACE_FLOWER
2: USE_AN_ACTION
3: VALID_MOVES

|: 1
FLOWER PLACEMENT
Which type of flower do you wish to place?
1: Purple: "P" (1 remaining)
2: Red: "R" (8 remaining)
3: Blue: "B" (1 remaining)
4: Yellow: "Y" (2 remaining)
5: White: "W" (4 remaining)
6: Green: "G" (2 remaining)

|: 1
Input coords to place a flower
|: 5:5
You scored 1 points this turn.

0 1 2 3 4 5 6 7 8 9 0
9 T
8
7
6 T
5
4
3
2
1
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |

Player 2 turn. You have completed 0 laps. Please select an option.
1: PLACE_FLOWER
2: USE_AN_ACTION
3: VALID_MOVES

|:

```

```
SICStus 4.3.3 (x86_64-win32-nt-4): Thu Jun 2 20:30:41 WEDT 2016
File Edit Flags Settings Help
1 2 3 4 5 6 7 8 9 0
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |
Player 3 input coords to place tree (Col:Row) and press [ENTER]
0 1 2 3 4 5 6 7 8 9 0
9 T 2 3 4 5 6 7 8 9 0
8 T
7 T
6 T
5 T
4 T
3 T
2 T
1 T
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |
Player 1 turn. You have completed 0 laps. Please select an option.
1 PLACE FLOWER
2 USE AN ACTION
3 VALID_MOVES
1
FLOWER PLACEMENT
Which type of flower do you wish to place?
1 Purple: "p" (1 remaining)
2 Red: "r" (8 remaining)
3 Blue: "b" (1 remaining)
4 Yellow: "y" (2 remaining)
5 White: "w" (4 remaining)
6 Green: "g" (2 remaining)
1
Input coords to place flower
0 1 2 3 4 5 6 7 8 9 0
9 T 2 3 4 5 6 7 8 9 0
8 T
7 T
6 T
5 p
4 T
3 T
2 T
1 T
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |
Player 2 turn. You have completed 0 laps. Please select an option.
1 PLACE FLOWER
2 USE AN ACTION
3 VALID_MOVES
1
MAKE AN ACTION
Which action do you want to use?
1 Swap a tree and a flower.
2 Swap two flowers.
3 Move a flower to an empty space.
4 Move a tree to an empty space.
5 Move your martian 5 spaces.
1
SWAP FLOWERS
Input coords of first flower
0 1 2 3 4 5 6 7 8 9 0
9 T 2 3 4 5 6 7 8 9 0
8 T
7 T
6 T
5 p
4 T
3 T
2 T
1 T
0 9 8 7 6 5 4 3 2 1 0
V | W | X | Y | Z |
```

5. Conclusões

Como conclusão, infelizmente temos de apontar que, o projeto foi levado de leve ânimo, pensando que iríamos ter tempo para o elaborar.

Infelizmente, organizamos mal o tempo e, no final fomos surpreendidos com algumas dificuldades e erros que nos custaram demasiado tempo, tornando impossível a elaboração íntegra do projeto.

Dentro das dificuldades, destacámos a dificuldade de fazer verificações, nomeadamente, quando queríamos verificar se uma posição já estava ocupada e se as coordenadas eram válidas. Dentro destas duas dificuldades, o maior desafio foi tratar das leituras que o utilizador introduzia (inputs) porque não conseguimos arranjar nenhuma maneira eficiente de tratar das leituras não desejadas.

Nem tudo foi mau, achámos interessante o desafio e o teste que este projeto nos impôs às capacidades de programação e adaptação e pretendemos levar a cabo a concretização do mesmo.

6. Bibliografia

6.1 Páginas web

Informações e imagens do jogo em:

http://nestorgames.com/rulebooks/GARDENSOFIO_EN.pdf

6.2 Documentação

Fornecida na página da unidade curricular do moodle.

7. Anexo

Segue em anexo, o código fonte.