

# Trabalho Prático CI1062

Pedro Amaral Chapelin, Vinicius Fontoura de Abreu, Heric Camargo

GRR20206145, GRR20206873, GRR20203959

## Introdução

Dividimos o trabalho em três partes principais:

- Board
- Entities
- Main

Cada uma implementando os conceitos de orientação a objetos, e **separando as responsabilidades de cada classe**.

Vamos detalhar cada uma das partes a seguir. Depois, vamos listar os requisitos do trabalho e como eles foram implementados.

## Desenvolvimento

### Board

- **Board class**
  - Criação do tabuleiro (uma matriz de setores)
  - criação das entidades
  - Criação dos itens
  - Movimentação das entidades
- **Sector class**
  - Estado do setor (vazio, restrito, ocupado com Player ou Fake News, com item)
  - **Coordenada** do setor

É o package responsável por **gerenciar o tabuleiro do jogo, e as entidades** que estão nele.

Nele está presente a classe Board, que é responsável por gerar o tabuleiro, e gerenciar as entidades. Ele também é responsável por gerenciar a interação entre as entidades.

### Entities

- **Entity class (Abstrasta)**

- Coordenada da entidade
- Estado da entidade (vivo, morto, adicionado recentemente ou removido do jogo)

É o package responsável por **implementar as entidades do jogo**, e seus métodos e atributos.

Temos 2 entidades principais, o jogador e a fake news. **Ambas herdam da classe abstrata Entity**, que contém os atributos e métodos comuns entre as entidades. Elas também **implementam a interface Movement**, que contém o método move, que é responsável por mover as entidades.

- **Player class**
  - Inventário do jogador
  - Tamanho máximo do inventário
  - Método para mover o jogador
  - Método para adicionar item ao inventário
- FakeNews class (Abstrata)
  - Método para adicionar uma fake news a um Setor

Existem 3 **Classes** de Fake News (F1,F2,F3) que herdam da classe abstrata e irão implementar seu próprio movimento e pode ter suas funções auxiliares.

- **Item (Abstrata)**
  - Coordenada do item
  - Método para usar o item (Abstrato)

Cada um dos 4 itens irá implementar seu próprio método de uso.

## Coordinate

- **Coordinate class**
  - Um par  $i, j$  que representa a coordenada de um setor
  - Métodos para calcular a distância entre duas coordenadas e conferir se coordenadas são válidas

## Cores

- Color class
  - Variáveis **estáticas** para as cores

## Main

É a classe principal do jogo, que contém o método **main**, e é responsável pelo **loop principal** do jogo e pela **interação** com o usuário.

Nela, dividimos o loop principal em 4 grandes partes:

- lógicas de vitória e derrota
- movimento do jogador

- usar item
- movimento das fake news

## Requisitos

Os requisitos indicados, e onde cada um foi implementado, foram:

1. **Classes (atributos e métodos)**
  - Board
  - Entity
  - Player
  - Fake News
  - Item
  - Cores
2. **Construtores**
  - **Todas** as classes possuem construtores
3. **Encapsulamento**
  - **Todas** as classes abstratas possuem atributos **privados**, e métodos públicos
4. **Herança**
  - Player e Fake News **herdam** de Entity
  - Itens **herdam** de Item
5. **Interface**
  - Players e Fake News **implementam a interface** Movement
6. **Classe Abstrata**
  - Entity
  - Fake News
  - Item
7. **Polimorfismo**
  - Polimorfismo de **Inclusão** no método move
  - Polimorfismo de **Coerção DownCast** nos métodos de ação de cada item e no movimento de cada fake news
8. **Coleção**
  - O inventário, Lista de Fake News, Lista de players e Lista de Itens são coleções são listas ligadas
  - Os métodos move de cada entidade utilizam HashMaps para mapear as direções

## Organização do desenvolvimento

Para evitar problemas e **seguir boas práticas de desenvolvimento de código**, utilizamos o **git** para versionar o código e, durante seu uso, um padrão de commits foi seguido da seguinte forma:

[<tipo>] <mensagem>

Onde <tipo> pode ser:



```

--> Player 1 turn
Press action desired [1 -> move / 2 -> use itens]
1
Turn 1 of 20
+-----+
0 |      |      |      | XX | J1 |      |      |      |
+-----+
1 |      |      |      |      |      |      | XX |      |
+-----+
2 | ?? |      | F3 |      |      | F2 | XX |      |
+-----+
3 |      |      |      |      |      |      |      |      |
+-----+
4 | J4 |      |      |      |      | F2 |      | F1 | J2 |
+-----+
5 |      |      |      |      |      |      |      |      |
+-----+
6 |      |      |      |      | F3 | XX | F1 |      |
+-----+
7 |      |      |      |      | ?? |      |      |      |
+-----+
8 |      |      |      |      | J3 |      |      |      |
+-----+
  0  1  2  3  4  5  6  7  8

Choose a direction to move [WASD]
s
Turn 1 of 20
+-----+
0 |      |      |      | XX |      |      |      |      |
+-----+
1 |      |      |      |      | J1 |      | XX |      |
+-----+
2 | ?? |      | F3 |      |      | F2 | XX |      |
+-----+
3 |      |      |      |      |      |      |      |      |
+-----+
4 | J4 |      |      |      |      | F2 |      | F1 | J2 |
+-----+
5 |      |      |      |      |      |      |      |      |
+-----+
6 |      |      |      |      | F3 | XX | F1 |      |
+-----+
7 |      |      |      |      | ?? |      |      |      |
+-----+
8 |      |      |      |      | J3 |      |      |      |
+-----+
  0  1  2  3  4  5  6  7  8

Player J1 moved to position: 1-4

```

Figure 2: Movimentação do jogador

```

Turn 1 of 20
  +---+---+---+---+---+---+---+---+
0 |   |   |   |   |   |   |   |   |   |
  +---+---+---+---+---+---+---+---+
1 |   |  ??  | F2 |   |   |   | XX | F3 |   |
  +---+---+---+---+---+---+---+---+
2 |   |   |   |   |   |   |   |   |   |
  +---+---+---+---+---+---+---+---+
3 |   |   | XX |   |   |   |   |   |   |
  +---+---+---+---+---+---+---+---+
4 | J4 |   |   |   |   |   |   |   | J2 |
  +---+---+---+---+---+---+---+---+
5 |   |   |   | F1 | ?? |   |   |   |   |
  +---+---+---+---+---+---+---+---+
6 |   |   | F3 | F1 |   |   |   |   |   |
  +---+---+---+---+---+---+---+---+
7 |   |   |   | XX |   | XX |   | F2 |   |
  +---+---+---+---+---+---+---+---+
8 |   |   |   |   | J3 |   |   |   |   |
  +---+---+---+---+---+---+---+---+
    0     1     2     3     4     5     6     7     8

Player J1 died! :(

```

Figure 3: jogador morrendo

```

Turn 4 of 20
+---+---+---+---+---+---+---+---+
0 | XX |   |   |   |   |   |   | J1 |   |
+---+---+---+---+---+---+---+---+
1 |   |   |   |   |   |   |   | XX |   |
+---+---+---+---+---+---+---+---+
2 |   |   |   | F3 |   |   |   |   |   |
+---+---+---+---+---+---+---+---+
3 |   |   |   |   |   |   | F1 |   |   |
+---+---+---+---+---+---+---+---+
4 |   |   | J4 | ?? |   |   |   |   |   |
+---+---+---+---+---+---+---+---+
5 |   |   | F1 |   |   |   |   |   |   |
+---+---+---+---+---+---+---+---+
6 |   | XX |   |   |   | J3 |   |   |   |
+---+---+---+---+---+---+---+---+
7 |   |   | XX |   |   |   |   |   |   |
+---+---+---+---+---+---+---+---+
8 |   |   |   |   |   |   | F2 | ?? |   |
+---+---+---+---+---+---+---+---+
  0   1   2   3   4   5   6   7   8

Player 1, this is your inventory
Choose the item you want to use, or 5 to return
=====INVENTORY=====

1 - Item Denunciar - Denunciar FN, elimina todas as FN ao redor do jogador.

```

Figure 4: menu de itens