

UNIVERSIDADE DA REGIÃO DE JOINVILLE
SISTEMAS DE INFORMAÇÃO

JOGO DE ESTRATÉGIA

MARIA FERNANDA FUCKNER

RICHARD W.M DE OLIVEIRA

CAMILE NOLLI

GUILHERME LUIS RAISER

JOÃO CAETANO GUIMARÃES

YDUAN RECH

JOÃO ENGEL

PROFESSOR PAULO

Fundamentos de Inteligência Artificial

Joinville – SC

2024

NOME DO JOGO ESCOLHIDO

Dominó

REGRAS DO JOGO

FORMA DE JOGAR

- **Início do Jogo:** O jogador e a máquina recebem 7 peças aleatórias do conjunto de dominó.
- **Turno do Jogador:** O jogador é solicitado a escolher uma peça da sua mão para jogar. O programa exibe as peças disponíveis na mão do jogador e solicita que ele selecione uma.
- **Turno da Máquina:** O programa avalia as peças disponíveis na mão da máquina e escolhe a peça mais adequada para jogar. A máquina, se tiver mais de uma opção para jogar, escolhe sempre a peça mais alta.
- **Validação de Jogada:** Após cada jogada, o programa verifica se a peça jogada pelo jogador ou pela máquina é válida de acordo com as regras do dominó. Caso não seja válida, lança um erro.
- **Continuação do Jogo:** Os turnos alternam entre jogador e máquina até que um dos jogadores fique sem peças na mão ou até que nenhum dos jogadores consiga jogar, tendo que comprar uma peça.
- **Fim do Jogo:** O jogo termina quando um dos jogadores não tem mais peças para jogar ou quando nenhum dos jogadores pode jogar uma peça. O programa então exibe o vencedor.

COMO O JOGO FOI DESENVOLVIDO

Linguagem de Programação e Ambiente de Desenvolvimento:

Desenvolvido no versátil Java, o jogo de Dominó tira proveito das inúmeras classes disponíveis que aprimoram a portabilidade e robustez das aplicações. Para a codificação, utilizou-se o Visual Studio Code, um editor multifuncional que, além de suportar várias linguagens, traz recursos como depuração e gerenciamento de Git.

Arquitetura e Design do Software:

Orientação a Objetos:

A arquitetura do software se baseia na orientação a objetos, simplificando tanto a organização quanto a manutenção do código. Classes representativas como Pedra, Jogador, JogadorHumano, JogadorIA e Tabuleiro são pilares deste modelo.

Encapsulamento:

A integridade do estado interno dos objetos é assegurada pelo encapsulamento, com a utilização criteriosa de acessores e mutadores para mediar as interações e modificações de dados entre as classes.

Interfaces:

Interfaces como a 'Jogador' estabelecem contratos essenciais para as classes JogadorHumano e JogadorIA, assegurando que ambos cumpram com os métodos necessários para a interação eficaz com o tabuleiro e o gerenciamento das pedras.

Gerenciamento de Estados:

A classe central Controle é a maestrina do jogo, gerindo os estados, como turnos e validações de jogadas, e assegurando a harmonia entre jogadores e o tabuleiro.

Funcionalidades Implementadas:

Geração e Embaralhamento de Pedras:

Cada partida inicia com a geração e o embaralhamento das pedras de dominó, garantindo equidade e aleatoriedade desde o começo.

Validação de Jogadas:

As jogadas passam por uma validação rigorosa para confirmar a compatibilidade das pedras com as regras do jogo, preservando a integridade do desafio.

Interface de Usuário:

A interface, embora simplista, oferece uma comunicação clara e opções de jogadas que facilitam a interação do usuário com o jogo.

Desenvolvimento Iterativo e Testes:

Adotando um método iterativo, o projeto evolui com testes contínuos, garantindo a funcionalidade de cada novo elemento antes de avançar, minimizando falhas e refinando o código final.