

Projeto 2 – Dominó em Dupla



Este projeto é para ser desenvolvido em dupla. Créditos: Projeto desenvolvido pelo Prof. Santiago V. Ravelo.

Introdução

O dominó em duplas é um jogo popular nas ruas de Havana, Cuba. A tradição deste jogo é tão profunda que até mesmo o Cemitério de Colón (Colombo) possui uma história singular: uma mulher que faleceu após perder uma partida de dominó, cujos filhos homenagearam sua paixão pelo jogo construindo seu túmulo em forma da peça com que ela perdeu (o (3,3)), incluindo a disposição da partida (segue uma foto).



Por causa dessa paixão, os habitantes da ilha decidiram fazer um torneio global para decidir as pessoas que teriam direito a explorar a recém descoberta cidade submersa da Atlântida, nos mares do arquipélago. Isto chamou a atenção de vários indivíduos, entre eles empresários avaros como Mr Burns, Mr Krabs e Scrooge McDuck, e pessoas pouco expertas como Homer Simpson, Patrick Star e Philip Fry, que se juntaram em duplas para tentar tomar posse da ilha. Os empresários, com sua mentalidade gananciosa, acabam refletindo isso no jogo, usando estratégias que priorizam jogar fora as peças de maior valor. Enquanto isso, os pouco expertos, conhecidos pela falta de intelecto e tomada de decisões arbitrárias e sem fundamento, adotam uma estratégia tola, que joga qualquer peça que seja possível colocar (ou seja, eles não têm estratégia).

Os habitantes da ilha, querem garantir uma exploração justa e completa da cidade perdida, preocupados com as duplas formadas, pedem a sua ajuda para criar uma dupla que consiga vencer as outras no torneio.

Seu objetivo é implementar uma estratégia vencedora para jogar em dupla contra outras estratégias. Você terá à sua disposição duas estratégias básicas para comparar: uma tola e outra gananciosa. Na primeira, é jogada qualquer peça das possíveis, o que basicamente equivale a não possuir estratégia. Na segunda, é jogada

a peça com maior valor entre as que se pode jogar, sendo esta uma estratégia avarenta que tenta se desfazer das peças mais custosas.

Seu desafio é criar uma dupla de jogadores cuja estratégia supere qualquer dupla formada por jogadores tolos ou gananciosos. No dia do torneio, serão realizadas 1000 partidas entre cada par de duplas para garantir a eficácia da estratégia, pois, nesse jogo, ocasionalmente uma estratégia ruim pode ganhar por questões de aleatoriedade.

No torneio, não só participarão as duplas acima, mas qualquer outra dupla inscrita. Será que vocês conseguem ganhar das duplas tolas ou gananciosas e garantir uma das três primeiras posições para ganhar o direito de explorar a Atlântida?

O projeto será avaliado em um torneio entre todas as duplas criadas, com premiação para as três primeiras colocadas.

O Jogo

- **Componentes do jogo:**

- Um conjunto de dominó, com peças que vão de (0,0) a (9,9), totalizando 55 peças.
- Quatro jogadores, formando duas duplas (parceiros sentam-se frente a frente).
- Um juiz.

- **Objetivo do jogo:**

- Vence a dupla que tiver um jogador sem peças primeiro.
- Se o jogo fechar e todos os jogadores ainda tiverem peças, vence a dupla com o jogador que somados os pontos das peças tiver menor valor.
- Caso os jogadores com menor valor em cada dupla tiverem a mesma soma de pontos, o jogo empata.

- **Início do jogo:**

- O juiz embaralha as peças.
- O juiz entrega a cada jogador 10 peças aleatórias. Os jogadores não sabem as peças dos outros, só as próprias.

- **Ordem de jogada:**

- No primeiro jogo entre duas duplas, o juiz escolhe um jogador aleatório para começar, nos seguintes, vai mudando o jogador em sentido horário.
- O jogador que começa, coloca a peça que desejar na mesa.
- As jogadas (turnos) seguem no sentido horário.

- **Desenvolvimento do jogo:**

- Cada jogador, em sua vez, deve colocar uma peça com um número correspondente a uma das extremidades das peças já jogadas.
- Se um jogador não puder jogar, não quiser jogar, ou tentar uma jogada inválida, é equivalente a perder o turno e continua com as mesmas peças.

- **Estratégias de jogo:**

- Colaboração: Parceiros devem tentar lembrar as peças jogadas e trabalhar juntos para bloquear os oponentes e ajudar um ao outro a se livrar de peças difíceis.
- Comunicação Silenciosa: Jogadores não podem falar sobre as peças, mas devem prestar atenção às jogadas dos parceiros para entender a estratégia.

- Controle de Extremidades: Jogar peças que controlem as extremidades do jogo pode ser crucial para forçar os oponentes a passar.

- **Fim do jogo:**

- O jogo termina quando um jogador joga sua última peça ou quando os quatro jogadores perderam seu turno de forma seguida.
- Se um jogador termina suas peças, sua dupla ganha a rodada.
- Senão, os pontos das peças que cada jogador ainda possui são contados, vence a dupla que tiver o jogador com menos pontos. Se nas duas duplas, o jogador com menos pontos tem o mesmo valor, as duplas empatam.

No confronto entre duas duplas elas jogarão 1000 jogos, ganha o confronto a dupla que vencer mais jogos, em caso das duas vencerem a mesma quantidade de jogos, então empatam.

A colocação final do torneio é decidida pelo seguinte critério:

1. Ganha a dupla com mais confrontos vencidos.
2. Em caso de empate, ganha a dupla com menos confrontos perdidos.
3. Em caso de empate, ganha a dupla com mais jogos vencidos.
4. Em caso de empate, ganha a dupla com menos jogos perdidos.
5. Em caso de empate, as duplas empatam.

O Código

Todo o código está dentro da pasta 'dominoes'. Vocês só precisam alterar o arquivo 'student_players.py'.

Segue uma breve descrição do que cada arquivo faz:

- `main.py`: Contém a entrada ao projeto. É este arquivo que deve ser chamado com: `python3 main.py`. Para mudar a velocidade dos turnos (em milissegundos) e o número de jogos no confronto entre duas duplas, pode usar respectivamente '-s' e '-n': `python3 main.py -s 100 -n 10` (esse chamado deixa o jogos com 100 milissegundos por turno e 10 jogos por cada confronto entre duplas).
- `tournament.py`: Contém as funções que realizarão o torneio.
- `judge.py`: Contém a classe Judge, que é o juiz em cada partida.
- `drawer.py`: Contém as funções, cores e posições que permitem o desenho gráfico do jogo.
- `basic_players`: Contém a implementação da classe Player (que vocês devem herdar para realizar a sua estratégia), adicionalmente contém duas herdeiras: DummyPlayer e GreedyPlayer, cada uma implementando uma estratégia diferente (a tola e a gananciosa).
- `student_players.py`: Este é o arquivo que irão modificar. Nele podem implementar até duas herdeiras de Player diferentes contendo a estratégia da dupla. Ademais, devem atualizar as funções 'pair_name', que deve retornar o nome que definiram para sua dupla; e 'create_pair', que deve retornar uma tupla com dois elementos (dos objetos representando os jogadores da sua dupla).

Vocês não precisam entender 100% desses outros arquivos, mas sintam-se à vontade para dar uma lida. (Não os altere, apenas o 'student_players.py')

Que informações usar?

- A classe `Player` contém uma propriedade `tiles`, esta é uma lista das peças que o jogador tem no momento. Essa lista é atualizada pelo juiz quando, no turno do jogador, ele faz uma jogada válida.
- É proibido “trapacear” tentando olhar as peças da dupla (ou seja, não podem criar variáveis globais, classes ou funções que permitam um jogador ter acesso às peças que não são dele!).
- No turno do jogador, o juiz irá chamar o método `play`, que vocês devem implementar na(s) classe(s) herdeira(s):
 - O método recebe dois parâmetros:
 - * `board_extremes`: Uma tupla (e,d) contendo o número no extremo esquerdo (e) no tabuleiro e no extremo direito (d). A peça que vocês jogarem deve coincidir um um desses extremos. Se a tupla for vazia, é porque essa será a primeira jogada.
 - * `play_hist`: Uma lista que contem o histórico das jogadas na ordem em que foram feitas. Cada jogada é por sua vez uma tupla com 4 valores (p, ext, side, tile):
 - p: A posição do jogador no tabuleiro (em cada jogo os jogadores tem uma posição entre 0 e 3). Jogadores da mesma dupla tem as posições 0 e 2, ou 1 e 3 (ou seja posições com igual paridade). Assim, os adversários terão posições com paridade diferente.
 - ext: Os extremos do tabuleiro no turno dessa jogada (antes de colocar a peça jogada).
 - side: O lado pelo qual o jogador decidiu fazer a jogada (0 para esquerdo, 1 para direito).
 - tile: A peça jogada (que é uma tupla com dois inteiros). Se o jogador tentou uma jogada inválida, não tinha peças, ou decidiu não jogar, então o valor é `None`.
 - O método retorna dois valores: `side`, `tile`:
 - * `side`: Deve indicar por qual lado o jogador deseja jogar: esquerda (0) e direita(1). Mesmo não podendo jogar deve informar um lado. Se faz uma jogada e informa o lado errado o juiz coloca no certo.
 - * `tile`: A peça a ser jogada (deve ser uma jogada válida, senão o juiz considera que perde o turno). Se não puder jogar, esse valor é `None`.

Regras

1. Vocês não devem alterar o código dos outros arquivos além do '`student_players.py`'. Vocês podem criar mais arquivos, caso prefiram, mas não é necessário.
2. A sua estratégia só pode usar informação baseada nas peças que o jogador tem. Ele não tem acesso às do parceiro na dupla, nem às dos jogadores das outras duplas. Adicionalmente, pode (deve) usar o histórico de jogadas para a sua estratégia.
3. No início de cada partida o jogador recebe um número (posição) entre 0 e 4 na mesa, esse número serve para identificar as jogadas (e a ordem em que serão feitas). Jogadores de uma mesma dupla, recebem posições com igual paridade. Após o confronto entre duas duplas (100 jogos) essas posições podem mudar.
4. Também no início de cada jogo, cada jogador recebe 10 peças aleatórias.
5. O jogo acaba se um jogador ficar sem peças (tiver colocado todas). O juiz controla isso. Também acaba se os quatro jogadores perderam o turno de forma seguida (controlado também pelo juiz).
6. Se um jogador fica sem peças, então sua dupla ganha o jogo. Senão, ganha a dupla com o jogador que tenha menos pontos nas peças que ficaram, se o valor for o mesmo nas duas duplas, então há empate.

Submissão

Vocês devem entregar seu projeto pelo Google Classroom, compactando toda a pasta em um único arquivo '.tar.xz' ou '.zip'.

- Vocês devem compactar a pasta do projeto e não somente o 'dominoes/'.
- Junto com seu código, você deve submeter um '.txt' explicando, **sucintamente**, a estratégia que tentou implementar.
- O nome do arquivo comprimido deve conter os RAs da dupla separados por um underline. Exemplo: '123456_654321.zip'.
- Performance esperada: O seu código não precisa ser o mais rápido do mundo, nem o mais preciso do mundo. Mas, ele ainda sim precisa ser melhor do que duplas formadas por jogadores tolos e/ou gananciosos.
- Esperamos que haja um "esforço genuíno" no desenvolvimento de alguma estratégia. Tem que haver alguma estratégia.
- O código deve aderir a todas as convenções, normas e boas práticas de código que vimos no curso até hoje.

Torneio

Quando todas entregas forem submetidas, iremos realizar um torneio entre todas as duplas. Cada dupla confrontará todas as outras. As quatro melhores duplas irão para a Semifinal. Essas quatro duplas serão agrupadas aleatoriamente em dois confrontos, as vencedoras disputarão a primeira e segunda posição em um confronto e as perdedoras a terceira posição.

Os prêmios são:

- 1º lugar: equivalente a nota máxima de 2 laboratórios.
- 2º lugar: equivalente a nota máxima de 1.5 laboratórios.
- 3º lugar: equivalente a nota máxima de 0.5 laboratório.

Além disso, os ganhadores receberão o maior prêmio de todos: um parabéns e aperto de mão.