Jogo da Velha Descrição

Iago Wesley Nogueira Cordeiro Keline Morais Balieiro

March 2019

1 Descrição das funções

1.1 main, SelecionaJogador e variáveis globais

Para realizar a inteligência deste jogo foi utilizada uma árvore de tomada de decisões onde cada jogada é um nível da árvore. Inicialmente utilizamos um vetor de 9 posições vazio para definir as 9 casas do jogo da velha, cada casa ao decorrer do jogo pode ter um espaço em branco definido pela variável global 'empt', ter um X onde é a jogada do jogador definido pela variável global x ou O, a jogada do computador e definido pela variável global 'o'. A seguir utilizando a função SelecionaJogador() definimos quem irá começar o jogo escolhendo um número aleatório entre 0 e 1000 e tirando mod 2, o jogador começa caso seja igual a 0 e o computador caso seja igual a 1. Então inicia-se o jogo:

1.2 Jogo

Começando o jogo primeiro imprime o tabuleiro utilizando a função Imprime() e utilizando o vetor tabuleiro onde marca as jogadas ja realizadas. Então é utilizada uma função para cada jogada realizada sendo 9 jogadas no total. A IA toma decisões nas jogadas de número impar caso comece o jogo e de número par caso o jogador comece o jogo.

1.3 Estrategia e estrategiav

A função estratégia é utilizada para verificar se há possibilidade do jogador ganhar na próxima rodada, através de uma lista de condições verifica se há duas jogadas do jogador e um espaço em branco em cada linha, coluna e diagonal. Caso haja essa possibilidade a IA faz a jogada no espaço em branco fazendo assim com que o jogador não ganhe na próxima rodada utilizando este espaço em branco onde a IA fez a jogada. Similarmente a função estrategiav faz a mesma verificação que a função estrategia com a diferença que agora verifica se há em uma linha, coluna ou diagonal duas jogadas feitas pela IA e um espaço em branco, fazendo com que jogar neste espaço em branco dê a vitória para a IA.

1.4 VerificaJogo

A função verifica jogo verifica se há algum vencedor no jogo, através de uma lista de condições busca se há alguma linha, coluna ou diagonal preenchida totalmente com X ou O, retornando se o vencedor caso aconteça ou retornando 0 caso ainda não haja vencedores.

1.5 Jogada1-9

Em cada função de jogada de 1 a 9 primeiro verifica-se se é a vez do jogador efetuar a jogada, caso sim o jogador efetua a jogada em uma casa vazia, caso escolha uma casa já preenchida é pedido para que ele escolha novamente e retorna para a função Jogo onde chama a próxima jogada ou encerra o jogo caso haja um vencedor ou não tenha mais jogadas possíveis. No caso de ser a vez da IA fazer a jogada, é feita a verificação de possibilidade do jogador ganhar na próxima jogada ou da IA ganhar com a jogada atual, caso aconteça a jogada é feita baseada nisso, utilizando as funções Estrategia e estrategiav, não sendo o caso a IA escolhe a próxima jogada possível através da lista de decisões.

2 Estratégia

2.1 A IA começa

A IA começando as tomadas de decisões são feitas nas funções de jogadas ímpares(1, 3, 5, 7 e 9)

2.1.1 Jogada 1

Iniciando a IA joga na primeira casa.

2.1.2 Jogada 3

Seguindo a estratégia a IA verifica se o jogador jogou na última casa, caso não a IA faz a jogada. Caso contrário a IA escolheria um dos outros dois cantos disponíveis e faria sua jogada, para simplificar foi escolhido o canto de baixo ou seja a sétima casa.

2.1.3 Jogada 5

Seguindo os passos da jogada anterior, se o jogador não tiver jogado última casa obrigatoriamente ele terá que jogar na casa do meio ou perde o jogo, sendo assim as próximas jogadas são decididas pelas funções Estrategia e estrategiav pois a partir deste ponto todas as possibilidades levam a vitória ou a derrota. Caso o jogador tenha escolhido a ultima casa em sua primeira jogada, obrigatoriamente ele deve jogar na sexta casa ou perderá, sendo assim a IA escolhe o canto superior ou seja a terceira casa, tendo assim a certeza da vitória em sua próxima jogada.

2.1.4 Jogada 7

Neste passo a jogada é feita exclusivamente pelas funções Estrategia e estrategiav pois não há mais opções de escolha que não levem a vitória ou a derrota.

2.1.5 jogada 9

Este passo é idêntico ao anterior, mas por ser a última jogada ou seja ter apenas uma casa vazia foi escolhida esta como jogada da IA.

2.2 O jogador começa

O jogador começando as tomadas de decisões são feitas nas funções de jogadas pares(2, 4, 6, 8)

2.2.1 Jogada 2

Como primeira jogada da IA é verificado se o jogador fez sua jogada na casa do meio, caso não esta será a jogada da IA, caso sim a IA escolhe a primeira casa como sua jogada.

2.2.2 Jogada 4

O jogador tendo jogado no meio em sua primeira jogada, sua segunda jogada define onde a IA irá jogar pois escolhendo o meio em primeiro lugar qualquer outra jogada que o jogador fizer faz com que sua próxima jogada seja a vitória caso a IA não jogue na casa vazia da linha, coluna ou diagonal preenchida com duas jogadas do jogador. Caso contrário a IA verifica se ainda há possibilidade de vitória: na coluna central e faz sua jogada na segunda casa, na linha central e faz sua jogada na quarta casa ou na diagonal principal e efetua sua jogada na primeira casa.

2.2.3 Jogada 6

Neste passo a IA escolhe um dos 6 estados possíveis para escolher caso não haja possibilidade do jogador ganhar na próxima jogada ou a IA na jogada atual, a escolha é feita evitando que haja janela para o jogador ganhar nas próximas jogadas.

2.2.4 Jogada 8

Neste passo não havendo possibilidade de vitória da IA, ou do jogador na próxima jogada a IA joga na primeira das duas casas vazias.

3 Codigo

```
#Created on Tue Mar 19 09:59:56 2019
#================================================================================
#============================JOGO DA VELHA=======================================
#================================================================================
#========================equipe: IAGO E KELINE===================================
#================================================================================

import random
import time
import os

x = 'X'
o = 'O'
empt = '_'
limpar = 'clear'

#fun   o para imprimir o tabuleiro na tela
def Imprime(raiz):
    print("tabuleiro__________mapa")
    print(raiz[0], "|", raiz[1],"|", raiz[2],"______1_|_2_|_3")
    print("----------______----------")
    print(raiz[3], "|", raiz[4],"|", raiz[5],"______4_|_5_|_6")
    print("----------______----------")
    print(raiz[6], "|", raiz[7],"|", raiz[8],"______7_|_8_|_9")


#fun   o que verifica se h  algum vencedor
def verificaJogo(tabuleiro):
    if(tabuleiro[0]==tabuleiro[3] and tabuleiro[3]==tabuleiro[6] and tabuleiro[0]!=empt):
        return tabuleiro[0]

    elif(tabuleiro[1]==tabuleiro[4] and tabuleiro[4]==tabuleiro[7] and tabuleiro[1]!=empt):
        return tabuleiro[1]

    elif(tabuleiro[2]==tabuleiro[5] and tabuleiro[5]==tabuleiro[8] and tabuleiro[2]!=empt):
            return tabuleiro[2]

    elif(tabuleiro[0]==tabuleiro[1] and tabuleiro[1]==tabuleiro[2] and tabuleiro[0]!=empt):
            return tabuleiro[0]

    elif(tabuleiro[3]==tabuleiro[4] and tabuleiro[4]==tabuleiro[5] and tabuleiro[3]!=empt):
            return tabuleiro[3]

    elif(tabuleiro[6]==tabuleiro[7] and tabuleiro[7]==tabuleiro[8] and tabuleiro[6]!=empt):
            return tabuleiro[6]

    elif(tabuleiro[0]==tabuleiro[4] and tabuleiro[4]==tabuleiro[8] and tabuleiro[0]!=empt):
            return tabuleiro[0]

    elif(tabuleiro[2]==tabuleiro[4] and tabuleiro[4]==tabuleiro[6] and tabuleiro[2]!=empt):
            return tabuleiro[2]
    else:
        return 0

#fun   o para definir se h  possibilidade do computador ganhar e fazer a jogada para vit ria
def Estrategiav(tabuleiro):
    if(tabuleiro[0]==o and tabuleiro[1]==o and tabuleiro[2]==empt):
        tabuleiro[2]=o
        return 1
    elif(tabuleiro[0]==o and tabuleiro[2]==o and tabuleiro[1]==empt):
        tabuleiro[1]=o
        return 1
    elif(tabuleiro[1]==o and tabuleiro[2]==o and tabuleiro[0]==empt):
        tabuleiro[0]=o
        return 1
    elif(tabuleiro[3]==o and tabuleiro[4]==o and tabuleiro[5]==empt):
        tabuleiro[5]=o
        return 1
    elif(tabuleiro[3]==o and tabuleiro[5]==o and tabuleiro[4]==empt):
        tabuleiro[4]=o
        return 1
    elif(tabuleiro[4]==o and tabuleiro[5]==o and tabuleiro[3]==empt):
        tabuleiro[3]=o
        return 1
    elif(tabuleiro[6]==o and tabuleiro[7]==o and tabuleiro[8]==empt):
        tabuleiro[8]=o
        return 1
    elif(tabuleiro[6]==o and tabuleiro[8]==o and tabuleiro[7]==empt):
        tabuleiro[7]=o
        return 1
    elif(tabuleiro[7]==o and tabuleiro[8]==o and tabuleiro[6]==empt):
        tabuleiro[6]=o
        return 1
    elif(tabuleiro[0]==o and tabuleiro[4]==o and tabuleiro[8]==empt):
        tabuleiro[8]=o
        return 1
    elif(tabuleiro[0]==o and tabuleiro[8]==o and tabuleiro[4]==empt):
```

```
            tabuleiro[4]=o
            return 1
        elif(tabuleiro[4]==o and tabuleiro[8]==o and tabuleiro[0]==empt):
            tabuleiro[0]=o
            return 1
        elif(tabuleiro[2]==o and tabuleiro[4]==o and tabuleiro[6]==empt):
            tabuleiro[6]=o
            return 1
        elif(tabuleiro[2]==o and tabuleiro[6]==o and tabuleiro[4]==empt):
            tabuleiro[4]=o
            return 1
        elif(tabuleiro[4]==o and tabuleiro[6]==o and tabuleiro[2]==empt):
            tabuleiro[2]=o
            return 1
        elif(tabuleiro[0]==o and tabuleiro[3]==o and tabuleiro[6]==empt):
            tabuleiro[6]=o
            return 1
        elif(tabuleiro[0]==o and tabuleiro[6]==o and tabuleiro[3]==empt):
            tabuleiro[3]=o
            return 1
        elif(tabuleiro[3]==o and tabuleiro[6]==o and tabuleiro[0]==empt):
            tabuleiro[0]=o
            return 1
        elif(tabuleiro[1]==o and tabuleiro[4]==o and tabuleiro[7]==empt):
            tabuleiro[7]=o
            return 1
        elif(tabuleiro[1]==o and tabuleiro[7]==o and tabuleiro[4]==empt):
            tabuleiro[4]=o
            return 1
        elif(tabuleiro[4]==o and tabuleiro[7]==o and tabuleiro[1]==empt):
            tabuleiro[1]=o
            return 1
        elif(tabuleiro[2]==o and tabuleiro[5]==o and tabuleiro[8]==empt):
            tabuleiro[8]=o
            return 1
        elif(tabuleiro[2]==o and tabuleiro[8]==o and tabuleiro[5]==empt):
            tabuleiro[5]=o
            return 1
        elif(tabuleiro[5]==o and tabuleiro[8]==o and tabuleiro[2]==empt):
            tabuleiro[2]=o
            return 1

        else:
            return 0


#fun   o para definir se h   possibilidade do jogador ganhar e fazer uma jogada para tentar impedir
def Estrategia(tabuleiro):
    if(tabuleiro[0]==x and tabuleiro[1]==x and tabuleiro[2]==empt):
        tabuleiro[2]=o
        return 1
    elif(tabuleiro[0]==x and tabuleiro[2]==x and tabuleiro[1]==empt):
        tabuleiro[1]=o
        return 1
    elif(tabuleiro[1]==x and tabuleiro[2]==x and tabuleiro[0]==empt):
        tabuleiro[0]=o
        return 1
    elif(tabuleiro[3]==x and tabuleiro[4]==x and tabuleiro[5]==empt):
        tabuleiro[5]=o
        return 1
    elif(tabuleiro[3]==x and tabuleiro[5]==x and tabuleiro[4]==empt):
        tabuleiro[4]=o
        return 1
    elif(tabuleiro[4]==x and tabuleiro[5]==x and tabuleiro[3]==empt):
        tabuleiro[3]=o
        return 1
    elif(tabuleiro[6]==x and tabuleiro[7]==x and tabuleiro[8]==empt):
        tabuleiro[8]=o
        return 1
    elif(tabuleiro[6]==x and tabuleiro[8]==x and tabuleiro[7]==empt):
        tabuleiro[7]=o
        return 1
    elif(tabuleiro[7]==x and tabuleiro[8]==x and tabuleiro[6]==empt):
        tabuleiro[6]=o
        return 1
    elif(tabuleiro[0]==x and tabuleiro[4]==x and tabuleiro[8]==empt):
        tabuleiro[8]=o
        return 1
    elif(tabuleiro[0]==x and tabuleiro[8]==x and tabuleiro[4]==empt):
        tabuleiro[4]=o
        return 1
    elif(tabuleiro[4]==x and tabuleiro[8]==x and tabuleiro[0]==empt):
        tabuleiro[0]=o
        return 1
    elif(tabuleiro[2]==x and tabuleiro[4]==x and tabuleiro[6]==empt):
        tabuleiro[6]=o
        return 1
    elif(tabuleiro[2]==x and tabuleiro[6]==x and tabuleiro[4]==empt):
        tabuleiro[4]=o
        return 1
    elif(tabuleiro[4]==x and tabuleiro[6]==x and tabuleiro[2]==empt):
```

```
            tabuleiro[2]=o
            return 1
        elif(tabuleiro[0]==x and tabuleiro[3]==x and tabuleiro[6]==empt):
            tabuleiro[6]=o
            return 1
        elif(tabuleiro[0]==x and tabuleiro[6]==x and tabuleiro[3]==empt):
            tabuleiro[3]=o
            return 1
        elif(tabuleiro[3]==x and tabuleiro[6]==x and tabuleiro[0]==empt):
            tabuleiro[0]=o
            return 1
        elif(tabuleiro[1]==x and tabuleiro[4]==x and tabuleiro[7]==empt):
            tabuleiro[7]=o
            return 1
        elif(tabuleiro[1]==x and tabuleiro[7]==x and tabuleiro[4]==empt):
            tabuleiro[4]=o
            return 1
        elif(tabuleiro[4]==x and tabuleiro[7]==x and tabuleiro[1]==empt):
            tabuleiro[1]=o
            return 1
        elif(tabuleiro[2]==x and tabuleiro[5]==x and tabuleiro[8]==empt):
            tabuleiro[8]=o
            return 1
        elif(tabuleiro[2]==x and tabuleiro[8]==x and tabuleiro[5]==empt):
            tabuleiro[5]=o
            return 1
        elif(tabuleiro[5]==x and tabuleiro[8]==x and tabuleiro[2]==empt):
            tabuleiro[2]=o
            return 1

        else:
            return 0

    def jogada1(tabuleiro, jogador):
        if(jogador == 0):
            num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
            while(tabuleiro[num-1]!=empt):
                num=int(input('casa ja preenchida, tente novamente: '))
            tabuleiro[num-1] = x
            Imprime(tabuleiro)
            jogador = 1
        else:
            tabuleiro[0]=o
            print("Computador jogando...")
            time.sleep(3)
            os.system(limpar)
            Imprime(tabuleiro)
            jogador = 0
        return jogador

    def jogada2(tabuleiro, jogador):
        if(jogador == 0):
            num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
            while(tabuleiro[num-1]!=empt):
                num=int(input('casa ja preenchida, tente novamente: '))
            tabuleiro[num-1] = x
            Imprime(tabuleiro)
            jogador = 1
        else:
            if(tabuleiro[4]==empt):
                tabuleiro[4]=o
            else:
                tabuleiro[0]=o
            print("Computador jogando...")
            time.sleep(3)
            os.system(limpar)
            Imprime(tabuleiro)
            jogador = 0
        return jogador

    def jogada3(tabuleiro, jogador):
        if(jogador == 0):
            num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
            while(tabuleiro[num-1]!=empt):
                num=int(input('casa ja preenchida, tente novamente: '))
            tabuleiro[num-1] = x
            Imprime(tabuleiro)
            jogador = 1
        else:
            if(tabuleiro[8]==empt):
                tabuleiro[8]=o
            else:
                tabuleiro[6]=o
            print("Computador jogando...")
            time.sleep(3)
            os.system(limpar)
            Imprime(tabuleiro)
            jogador = 0
        return jogador

    def jogada4(tabuleiro, jogador):
```

```
    if(jogador == 0):
        num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
        while(tabuleiro[num-1]!=empt):
            num=int(input('casa ja preenchida, tente novamente: '))
        tabuleiro[num-1] = x
        Imprime(tabuleiro)
        jogador = 1
    else:
        if(Estrategia(tabuleiro)==0):
            if(tabuleiro[1]==empt and tabuleiro[7]==empt):
                tabuleiro[1]=o
            elif(tabuleiro[3]==empt and tabuleiro[5]==empt):
                tabuleiro[3]=o
            elif(tabuleiro[0]==empt and tabuleiro[8]==empt):
                tabuleiro[0]=o
        print("Computador jogando...")
        time.sleep(3)
        os.system(limpar)
        Imprime(tabuleiro)
        jogador = 0
    return jogador

def jogada5(tabuleiro, jogador):
    if(jogador == 0):
        num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
        while(tabuleiro[num-1]!=empt):
            num=int(input('casa ja preenchida, tente novamente: '))
        tabuleiro[num-1] = x
        Imprime(tabuleiro)
        jogador = 1
    else:
        if(Estrategiav(tabuleiro)==0):
            if(Estrategia(tabuleiro)==0):
                tabuleiro[2]=o
        print("Computador jogando...")
        time.sleep(3)
        os.system(limpar)
        Imprime(tabuleiro)
        jogador = 0
    return jogador

def jogada6(tabuleiro, jogador):
    if(jogador == 0):
        num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
        while(tabuleiro[num-1]!=empt):
            num=int(input('casa ja preenchida, tente novamente: '))
        tabuleiro[num-1] = x
        Imprime(tabuleiro)
        jogador = 1
    else:
        if(Estrategiav(tabuleiro)==0):
            if(Estrategia(tabuleiro)==0):
                if(tabuleiro==[o, empt, empt, x, x, o, empt, empt, x]):
                    tabuleiro[2]=o
                if(tabuleiro==[o, x, empt, x, o, empt, empt, empt, x]):
                    tabuleiro[2]=o
                if(tabuleiro==[empt, o, x, x, o, empt, empt, x, empt]):
                    tabuleiro[8]=o
                if(tabuleiro==[empt, o, empt, x, o, x, empt, x, empt]):
                    tabuleiro[0]=o
                if(tabuleiro==[x, o, empt, empt, o, x, empt, x, empt]):
                    tabuleiro[6]=o
                if(tabuleiro==[o, empt, empt, o, o, x, empt, x, empt]):
                    tabuleiro[2]=o
                elif(tabuleiro[1]==empt):
                    tabuleiro[1]=o
                elif(tabuleiro[3]==empt):
                    tabuleiro[3]=o
                elif(tabuleiro[5]==empt):
                    tabuleiro[5]=o
                elif(tabuleiro[6]==empt):
                    tabuleiro[6]=o
        print("Computador jogando...")
        time.sleep(3)
        os.system(limpar)
        Imprime(tabuleiro)
        jogador = 0
    return jogador


def jogada7(tabuleiro, jogador):
    if(jogador == 0):
        num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
        while(tabuleiro[num-1]!=empt):
            num=int(input('casa ja preenchida, tente novamente: '))
        tabuleiro[num-1] = x
        Imprime(tabuleiro)
        jogador = 1
    else:
        if(Estrategiav(tabuleiro)==0):
            print("Computador jogando...")
```

```
            if(Estrategia(tabuleiro)==0):
                print("Computador jogando...")
            time.sleep(3)
            os.system(limpar)
            Imprime(tabuleiro)
            jogador = 0
        return jogador

    def jogada8(tabuleiro, jogador):
        if(jogador == 0):
            num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
            while(tabuleiro[num-1]!=empt):
                num=int(input('casa ja preenchida, tente novamente: '))
            tabuleiro[num-1] = x
            Imprime(tabuleiro)
            jogador = 1
        else:
            if(Estrategiav(tabuleiro)==0):
                if(Estrategia(tabuleiro)==0):
                    if(tabuleiro[0]==empt):
                        tabuleiro[0]=o
                    elif(tabuleiro[1]==empt):
                        tabuleiro[1]=o
                    elif(tabuleiro[2]==empt):
                        tabuleiro[2]=o
                    elif(tabuleiro[3]==empt):
                        tabuleiro[3]=o
                    elif(tabuleiro[4]==empt):
                        tabuleiro[4]=o
                    elif(tabuleiro[5]==empt):
                        tabuleiro[5]=o
                    elif(tabuleiro[6]==empt):
                        tabuleiro[6]=o
                    elif(tabuleiro[7]==empt):
                        tabuleiro[7]=o
                    elif(tabuleiro[8]==empt):
                        tabuleiro[8]=o
            print("Computador jogando...")
            time.sleep(3)
            os.system(limpar)
            Imprime(tabuleiro)
            jogador = 0
        return jogador

    def jogada9(tabuleiro, jogador):
        if(jogador == 0):
            num=int(input('digite a casa que quer jogar de acordo com o mapa ao lado do tabuleiro: '))
            while(tabuleiro[num-1]!=empt):
                num=int(input('casa ja preenchida, tente novamente: '))
            tabuleiro[num-1] = x
            Imprime(tabuleiro)
            jogador = 1
        else:
            if(tabuleiro[0]==empt):
                tabuleiro[0]=o
            elif(tabuleiro[1]==empt):
                tabuleiro[1]=o
            elif(tabuleiro[2]==empt):
                tabuleiro[2]=o
            elif(tabuleiro[3]==empt):
                tabuleiro[3]=o
            elif(tabuleiro[4]==empt):
                tabuleiro[4]=o
            elif(tabuleiro[5]==empt):
                tabuleiro[5]=o
            elif(tabuleiro[6]==empt):
                tabuleiro[6]=o
            elif(tabuleiro[7]==empt):
                tabuleiro[7]=o
            elif(tabuleiro[8]==empt):
                tabuleiro[8]=o
            print("Computador jogando...")
            time.sleep(3)
            os.system(limpar)
            Imprime(tabuleiro)
            jogador = 0
        return jogador

    def Jogo(tabuleiro, jogador):
        Imprime(tabuleiro)
        vencedor=0
        jogador = jogada1(tabuleiro, jogador)
        jogador = jogada2(tabuleiro, jogador)
        jogador = jogada3(tabuleiro, jogador)
        jogador = jogada4(tabuleiro, jogador)
        jogador = jogada5(tabuleiro, jogador)
        vencedor=verificaJogo(tabuleiro)
        if(vencedor==0):
            jogador = jogada6(tabuleiro, jogador)
        vencedor=verificaJogo(tabuleiro)
        if(vencedor==0):
```

```
            jogador = jogada7(tabuleiro, jogador)
        vencedor=verificaJogo(tabuleiro)
        if(vencedor==0):
            jogador = jogada8(tabuleiro, jogador)
        vencedor=verificaJogo(tabuleiro)
        if(vencedor==0):
            jogador = jogada9(tabuleiro, jogador)
        vencedor = verificaJogo(tabuleiro)

        if(vencedor == x):
            print("O jogador venceu!")
        elif(vencedor==o):
            print("O computador venceu!")
        else:
            print("Empate!")

def SelecionaJogador():
    x=random.randint(0, 1000)
    return x % 2

def main():
    A = [empt] * 9  #Gera o tabuleiro vazio
    jogador = SelecionaJogador() #Seleciona qual jogador ir   come ar
    Jogo(A, jogador) #Inicia o jogo

main()
```