class Stone

def \_\_init\_\_(self, team):

self.team = team

self.wxP = [0,0]

self.wxM = [0,0]

self.wyP = [0,0]

self.wyM = [0,0]

self.wxPyP = [0,0]

self.wxPyM = [0,0]

self.wxMyP = [0,0]

self.wxMyM = [0,0]

self.bxP = [0,0]

self.bxM = [0,0]

self.byP = [0,0]

self.byM = [0,0]

self.bxPyP = [0,0]

self.bxPyM = [0,0]

self.bxMyP = [0,0]

self.bxMyM = [0,0]

self.priority = [[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0],[0,0]]

def priorityList(self):

if self.team == "White":

self.priority = [self.wxP, self.wxM, self.wyP, self.wyM, self.wxPyP, self.wxPyM, self.wxMyP, self.wxMyM]

elif self.team == "Black":

self.priority = [self.bxP, self.bxM, self.byP, self.byM, self.bxPyP, self.bxPyM, self.bxMyP, self.bxMyM]

return self.priority

def addStone(self, xPos, yPos,color):

stone = Stone(color)

self.stonesOrder.append([xPos,yPos,stone])

self.undoList.append(self.stones[xPos][yPos])

self.stones[xPos][yPos] = stone

if color == "White":

for i in range(1,5):

if self.stones[min(xPos+i, 15)][yPos].team == color:

if xPos+i < 16:

self.stones[min(xPos+i, 15)][yPos].wxM[0] += 5 - i

self.stones[xPos][yPos].wxP[0] += 5 - i

elif self.stones[min(xPos+i, 15)][yPos].team == 0:

self.stones[min(xPos+i, 15)][yPos].wxM[0] += 5 - i

else:

self.stones[min(xPos+i, 15)][yPos].wxM[1] = self.stones[min(xPos+i, 15)][yPos].wxM[0]

self.stones[xPos][yPos].wxP[1] = self.stones[xPos][yPos].wxP[0]

self.stones[min(xPos+i, 15)][yPos].wxM[0] = 0

self.stones[xPos][yPos].wxP[0] = 0

for i in range(1,5):

if xPos == 0:

donothing = 0

elif self.stones[max(xPos-i, 0)][yPos].team == color:

if xPos-i > -1:

self.stones[max(xPos-i, 0)][yPos].wxP[0] += 5 - i

self.stones[xPos][yPos].wxM[0] += 5 - i

elif self.stones[max(xPos-i, 0)][yPos].team == 0:

if xPos-i != 0:

self.stones[max(xPos-i, 0)][yPos].wxP[0] += 5 - i