class MaTran:

def \_\_init\_\_(self, ten\_tap\_tin):

with open(ten\_tap\_tin, 'r') as file:

self.hang, self.cot = map(int, file.readline().split())

self.luoi = [list(map(int, file.readline().split())) for \_ in range(self.hang)]

class Robot:

def \_\_init\_\_(self, x, y):

self.x, self.y = x, y

self.diem = 0

self.duong\_di = [(x, y)]

def co\_the\_di\_chuyen(self, x, y, ma\_tran):

return 0 <= x < ma\_tran.hang and 0 <= y < ma\_tran.cot and ma\_tran.luoi[x][y] > 0

def di\_chuyen(self, ma\_tran):

huong = [(0, -1), (0, 1), (-1, 0), (1, 0)]

x\_toi\_uu, y\_toi\_uu, gia\_tri\_max = -1, -1, -1

for dx, dy in huong:

x\_moi, y\_moi = self.x + dx, self.y + dy

if self.co\_the\_di\_chuyen(x\_moi, y\_moi, ma\_tran) and ma\_tran.luoi[x\_moi][y\_moi] > gia\_tri\_max:

x\_toi\_uu, y\_toi\_uu, gia\_tri\_max = x\_moi, y\_moi, ma\_tran.luoi[x\_moi][y\_moi]

if x\_toi\_uu != -1:

self.diem += ma\_tran.luoi[x\_toi\_uu][y\_toi\_uu]

ma\_tran.luoi[x\_toi\_uu][y\_toi\_uu] = 0

self.x, self.y = x\_toi\_uu, y\_toi\_uu

self.duong\_di.append((self.x, self.y))

class TroChoi:

def \_\_init\_\_(self, ten\_tap\_tin, hai\_robot=False):

self.ma\_tran = MaTran(ten\_tap\_tin)

self.robot1 = Robot(0, 0)

self.robot2 = Robot(self.ma\_tran.hang - 1, self.ma\_tran.cot - 1) if hai\_robot else None

self.hai\_robot = hai\_robot

def choi(self):

while True:

diem\_truoc\_1 = self.robot1.diem

self.robot1.di\_chuyen(self.ma\_tran)

if self.hai\_robot:

diem\_truoc\_2 = self.robot2.diem

self.robot2.di\_chuyen(self.ma\_tran)

if diem\_truoc\_1 == self.robot1.diem and diem\_truoc\_2 == self.robot2.diem:

break

elif diem\_truoc\_1 == self.robot1.diem:

break

def in\_ket\_qua(self):

print(self.robot1.diem)

print(" ".join(f"{x} {y}" for x, y in self.robot1.duong\_di))

if self.hai\_robot:

print(self.robot2.diem)

print(" ".join(f"{x} {y}" for x, y in self.robot2.duong\_di))

if \_\_name\_\_ == "\_\_main\_\_":

tro\_choi = TroChoi("input.txt", hai\_robot=True)

tro\_choi.choi()

tro\_choi.in\_ket\_qua()