import math

import random

def mm(pos, pts, d, a, b, is\_max):

if d == 0:

return pts[pos]

if is\_max:

max\_eval = -math.inf

for i in range(2):

n\_eval = mm(pos \* 2 + i, pts, d - 1, a, b, False)

max\_eval = max(max\_eval, n\_eval)

a = max(a, n\_eval)

if b <= a:

break

return max\_eval

else:

min\_eval = math.inf

for i in range(2):

n\_eval = mm(pos \* 2 + i, pts, d - 1, a, b, True)

min\_eval = min(min\_eval, n\_eval)

b = min(b, n\_eval)

if b <= a:

break

return min\_eval

def find\_winner(sid):

min\_pts = int(sid[4])

win\_pts = int(sid[-1:-3:-1])

max\_pts = int(win\_pts \* 1.5)

num\_shuffles = int(sid[3])

rnd\_pts\_list = random.sample(range(min\_pts, max\_pts), 8)

shf\_max\_pts\_list, count = [], 0

print(f'8 random points: {rnd\_pts\_list}\nTotal points to win: {win\_pts}')

result = mm(0, rnd\_pts\_list, 3, -math.inf, math.inf, True)

print(f'Alpha-beta pruning result = {result}')

if result >= win\_pts:

print('Optimus Prime wins')

else:

print('Megatron wins')

print('\nAfter shuffle:')

for \_ in range(num\_shuffles):

rnd\_pts\_list = random.sample(range(min\_pts, max\_pts), 8)

max\_val = max(rnd\_pts\_list)

shf\_max\_pts\_list.append(max\_val)

if max\_val >= win\_pts:

count += 1

shf\_max\_pt = max(shf\_max\_pts\_list)

print(f'List of all points values from each shuffle: {shf\_max\_pts\_list}\nThe maximum value of all shuffles: {shf\_max\_pt}\nWon {count} times out of {num\_shuffles} number of shuffles')

student\_id = '21301380'.replace('0', '8')

find\_winner(student\_id)