for i in range(len(lst)):

for j in range(i+1, len(Ist)):

[0,1,2,3]

i = 0, j = 1, 2 ,3

i = 1, j = 2, 3

...

for i in range(len(lst)):

for j in range(i):

i = 1, j = 0

i = 2; j = 0, 1

i = 3; j = 0, 1, 2

def Pick(player, brick, pile

choices.index(human\_decision[0])

ceil(x/y)

(x+y-1)/y

try:

human\_decision = input()

finally:

while len(discard) > 0:

main\_pile.append(discard.pop(-1))

deal\_computer = True

while len(main\_pile)>40:

if deal\_computer:

computer\_initial\_tower.append(main\_pile.pop(0))

deal\_computer = False

else:

human\_initial\_tower.append(main\_pile.pop(0))

deal\_computer = True

while len(main\_pile)>40:

main\_pile.pop(0)

[[2,3,4,0,-1,2]

[-5,2,1,7,3,0]

[0,2,1,4,1,5]]

lambda expense : 0

def func\_name(expense):

return 0

class A(object):

def \_\_str\_\_(self):

return 'test'

a = A()

print(a)