­­­­­­­­­class Link:

empty = ()

def \_\_init\_\_(self, first, rest=empty):

assert rest is Link.empty or isinstance(rest, Link)

self.first = first

self.rest = rest

def \_\_repr\_\_(self):

if self.rest is not Link.empty:

rest\_repr = ', ' + repr(self.rest)

else:

rest\_repr = ''

return 'Link(' + repr(self.first) + rest\_repr + ')'

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

string = '<'

while self.rest is not Link.empty:

string += str(self.first) + ' '

self = self.rest

return string + str(self.first) + '>'

输入以下指令，观察和自己预估是否一致。

>>> link = Link(1000)

>>> link.first

\_\_\_\_\_\_

>>> link.rest is Link.empty

\_\_\_\_\_\_

>>> link = Link(1000, 2000)

\_\_\_\_\_\_

>>> link = Link(1000, Link())

\_\_\_\_\_\_

>>> link = Link(1, Link(2, Link(3)))

>>> link.first

\_\_\_\_\_\_

>>> link.rest.first

\_\_\_\_\_\_

>>> link.rest.rest.rest is Link.empty

\_\_\_\_\_\_

>>> link.first = 9001

>>> link.first

\_\_\_\_\_\_

>>> link.rest = link.rest.rest

>>> link.rest.first

\_\_\_\_\_\_

>>> link = Link(1)

>>> link.rest = link

>>> link.rest.rest.rest.rest.first

\_\_\_\_\_\_

>>> link = Link(2, Link(3, Link(4)))

>>> link2 = Link(1, link)

>>> link2.first

\_\_\_\_\_\_

>>> link2.rest.first

\_\_\_\_\_\_