1. Given a 32-bit signed integer, reverse digits of an integer.

class Solution(object):

def reverse(self, x):

"""

:type x: int

:rtype: int

"""

a=str(x)

rev=''

if not(a.startswith('-')):

if a=='0':

rev='0'

else:

rev=a[::-1]

while(rev.startswith('0')):

rev=rev[1:]

else:

a=a[1:]

rev=a[::-1]

while(rev.startswith('0')):

rev=rev[1:]

rev='-'+rev

re=int(rev)

val1=re&-0x80000000

val2=re&0x7fffffff

print(val2)

if(val2==re) or (val1==-0x80000000):

return re

else:

return 0

1. Determine whether an integer is a palindrome. An integer is a palindrome when it reads the same backward as forward.

class Solution(object):

def isPalindrome(self, x):

"""

:type x: int

:rtype: bool

"""

num = bool

numstr = str(x)

if numstr.startswith('-'):

# numrev=numstr[::-1]

num = False

elif numstr.endswith('0') and len(numstr) != 1:

# numrev=numstr[::-1]

num = False

elif len(numstr) == 1:

num = True

elif numstr == '0' and len(numstr) == 1:

num = True

else:

numrev = numstr[::-1]

#print(numrev)

if numrev == numstr:

num = True

else:

num=False

return num