1.

class Solution:

def isPalindrome(self, x: int) -> bool:

range\_ = range(-2\*\*31, 2\*\*31-1)

if math.copysign(1,x) == 1:

y = str(x)

if str(x) == y[::-1]:

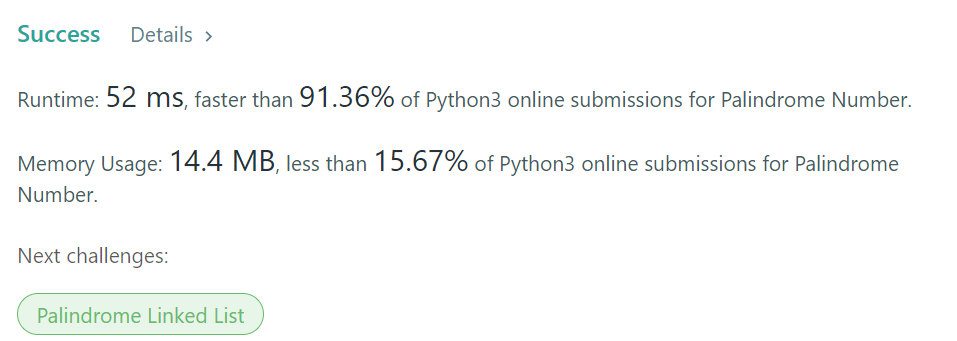
return True if x in range\_ else False

else:

return False

elif math.copysign(1,x) == -1:

return False



2. No need to confirm whether 負數

class Solution:

def isPalindrome(self, x: int) -> bool:

range\_ = range(-2\*\*31, 2\*\*31-1)

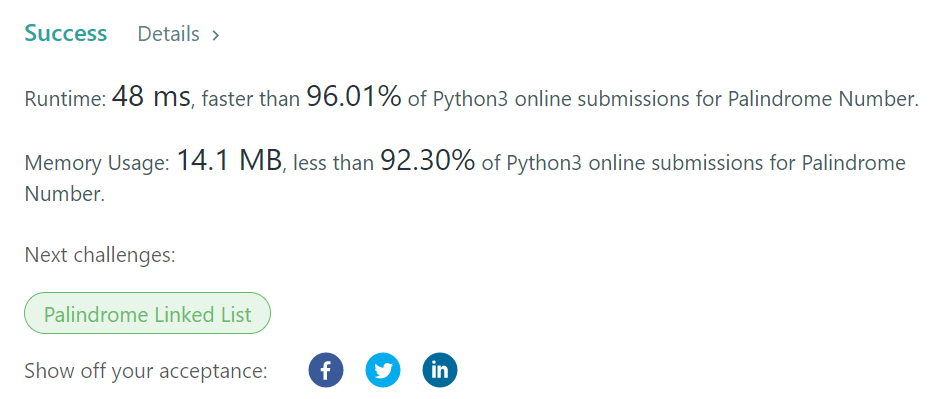
y = str(x)

if str(x) == y[::-1]:

return True if x in range\_ else False

else:

return False



java版本 x==rev/10 當x是長度是奇數會用到

public boolean isPalindrome(int x) {

if (x<0 || (x!=0 && x%10==0)) return false;

int rev = 0;

while (x>rev){

rev = rev\*10 + x%10;

x = x/10;

}

return (x==rev || x==rev/10);

}

同樣思路in python不轉成string

def isPalindrome(self, x: int) -> bool:

if x<0:

return False

inputNum = x

newNum = 0

while x>0:

newNum = newNum \* 10 + x%10

x = x//10

return newNum == inputNum

