**Justin\_Design\_0604.**  **Design Compressed String Iterator**

**Concept:**

collections.deque() ==> 兩頭的 lis，

有.popleft() 可以使用 ==> 刪除第一個，

有.appendleft()可以使用 ==> 從左邊(第一個)加入

re.findall(r'\D\d+') ==> 小 d 表示取數字 0-9，大 D 表示不要數字

# 注意數字 > 10

**Code:**

class StringIterator:

def \_\_init\_\_(self, compressedString: str):

self.value\_list = collections.deque()

string\_value\_list = re.findall(r'\D\d+', compressedString)

for value in string\_value\_list:

self.value\_list.append((value[0], int(value[1:])))

#print(string\_value\_list)

#print(self.value\_list)

def next(self) -> str:

temp\_value = ' '

if not self.value\_list:

return temp\_value

temp\_value, count = self.value\_list.popleft()

if count > 1:

self.value\_list.appendleft((temp\_value, count - 1))

return temp\_value

def hasNext(self) -> bool:

if self.value\_list:

return True

else:

return False

# Your StringIterator object will be instantiated and called as such:

# obj = StringIterator(compressedString)

# param\_1 = obj.next()

# param\_2 = obj.hasNext()