import csv

# Lists to store your csv columns

A = []

B = []

with open('health.csv') as f: #read your csv

reader = csv.reader(f, delimiter=',')

for row in reader: #appending the values

if row:

if row[0].isdigit():

A.append(int(row[0]))

if row[1].isdigit():

B.append(int(row[1]))

n = len(A) + 2 #n to calculate actual sum

def findMissing(A,B,n):

a\_sum, b\_sum = sum(A), sum(B) # O(2n)

actual\_sum = n\*(n+1)//2 # O(n)

b\_missing = actual\_sum - b\_sum

# Result 1 - integer that exists in B but not in A , // Part A

ret1 = actual\_sum - a\_sum - b\_missing

# Result2 - integer that is missing from both A and B, // Part B

ret2 = b\_missing

return ret1,ret2

print(findMissing(A,B,n))

# Part C

Yes, Part B is already been executed with the following:

1) space complexity: O(1) as the function uses variables to store the sum of 2 columns and they are independent of size of 2 columns.

2) time complexity : O(N), is when calculating the actual sum of 2 columns.