Problem # 1103 : Distribute Candies to People

<https://leetcode.com/problems/distribute-candies-to-people/>

Python 3 Solutions — Brute Force solution and improved solution

My first approach to solving this problem was the brute force method. My runtime for the Brute Force solution beats 73.32%. Then I did some improvement on this method and this improved method runtime beats 98.59%.

Method 1: Brute Force Solution: The candies are distributed to each person in turn and the number of candies left is calculated.

class Solution:

def distributeCandies(self, candies: int, num\_people: int) -> List[int]:

index = 0

arr = [0] \* num\_people

while candies > 0:

i = index % num\_people # i th person

if candies >= index + 1: # candies left is greater than what the i th person's share

arr[i] += index + 1 # give the i th person his/her share

candies = candies - (index + 1) # candies remaining

index += 1 # increment the index

else: # candies < index + 1 i.e. there are fewer candies than (index + 1)

arr[i] += candies # given the remaining candies to the i th person

candies = 0 # no candy is left as we gave away the last candies to the i th person

return(arr)

Method 2 : Improved method:

1. This is what is done in the first while loop. Calculate how many whole (or full) rounds of candy where every person gets the full share. Then there will the last round where some people will get candy and some may not and again a person getting the candy in the last round may not get his/her full share. The last round of candy is distributed to each person here in the else statement.
2. In the last for loop, the candies are distributed for the whole rounds, i.e. for all the rounds except the last one.

The i th person gets candies as follows: 1st round : i + 1 2nd round : num\_people + i + 1 3rd round : (2 \* num\_people) + i + 1 4th round : (3 \* num\_people) + i + 1 ... rth round: ((r - 1) \* num\_people) + i + 1 Total candies that the ith person gets in r full rounds = r \* (i + 1) + num\_people \* ( 1 + 2 + ... + (r-1))

= r \* (i + 1) + num\_people \* (r \*(r-1)/2)

I have used the formula that the sum of the first n natural numbers = n \* (n + 1) /2

My runtime for this solution beats 95t.94%

import math

class Solution:

def distributeCandies(self, candies: int, num\_people: int) -> List[int]:

arr = [0] \* num\_people

rounds = 0

while candies > 0:

candies4WholeRound = int((num\_people \* (num\_people + 1))/2 + (num\_people \*num\_people)\*rounds)

if candies >= candies4WholeRound:

candies -= candies4WholeRound

rounds += 1

else: # distribute the candies to each person until candies run out-- last round

for i in range(num\_people):

if candies >= num\_people \* rounds + i + 1:

arr[i] += num\_people \* rounds + i + 1

candies -= num\_people \* rounds + i + 1

else:

arr[i] += candies

candies = 0

# Now distribute the whole rounds of candies

for i in range(num\_people):

arr[i] += int((i+1)\*rounds + num\_people \* rounds \* (rounds - 1)/2)

return(arr)